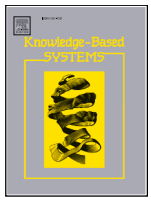
Knowledge-Based Systems 342 (2026) 115825

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys



Robust log-based multi-label feature selection with dynamic label correlation and relevance-redundancy optimization

Mohammad Faraji [a], Amjad Seyedi [b], Fardin Akhlaghian Tab [a,*]

[a] Department of Computer Engineering, University of Kurdistan, Iran
[b] Department of Mathematics and Operational Research, University of Mons, Belgium

ARTICLE INFO

Keywords:
Feature selection
Multi-label learning
Label correlation
Robustness
Sparseness

ABSTRACT

High-dimensional multi-label datasets commonly suffer from noisy annotations and redundant or irrelevant features, which jointly degrade learning accuracy and generalization. Although existing multi-label feature selection methods often adopt the $L_{2,1}$-norm to enhance robustness and induce sparsity, this formulation remains sensitive to extreme noise, relies on static label correlation modeling, and inadequately controls redundancy among selected features. To address these limitations, we propose Robust Log-based Multi-Label Feature Selection with Dynamic Label Correlation and Relevance–Redundancy Optimization (RLBMLFS). The proposed method introduces an element-wise logarithmic robust loss that effectively suppresses the influence of large reconstruction errors, providing stronger resilience to noisy labels than conventional sample-wise losses. In addition, RLBMLFS learns label dependencies dynamically from reconstructed labels, yielding a stable and adaptive label correlation structure that mitigates noise propagation. To jointly promote sparsity and reduce feature redundancy, we further incorporate a logarithmic redundancy-aware penalty together with an $L_{2,\log}$ pseudo-norm regularization, which offers a closer approximation to L_0-norm sparsity while alleviating the dominance of large feature weights. Extensive experiments on 17 real-world multi-label datasets across five evaluation metrics demonstrate that RLBMLFS consistently outperforms state-of-the-art methods in terms of robustness, sparsity quality, and classification performance. The source code is publicly available at: https://github.com/FarajiMohammad/RLBMLFS.

1. Introduction

Machine learning is widely used to solve complex problems across different fields such as natural language processing, computer vision, and bioinformatics. These applications often involve vast amounts of high-dimensional data. As data grows, multi-label datasets have emerged, where each instance can belong to multiple categories or labels at the same time. For instance, real-world entities often exhibit multiple labels concurrently. In the field of genomics, a single gene can carry out diverse functions, including photosynthesis, protein degradation, and signal transduction [1]. Multi-label data often involves high-dimensional feature spaces and a large number of labels, both of which are susceptible to noise. Dealing with high-dimensional data presents several challenges, such as increased learning complexity, higher computational resource demands, and reduced classification accuracy, often referred to as the curse of dimensionality. To address these challenges, feature selection methods have been developed to reduce the number of features by identifying and selecting the most relevant ones [2,3].

Feature selection techniques are generally categorized into three main types: filter, wrapper, and embedded methods. Filter methods operate independently of learning algorithms by ranking features before classification by evaluating data characteristics such as mutual information and correlation coefficients [4]. Wrapper methods, on the other hand, depend on learning algorithms and often employ evolutionary techniques to identify the optimal subset of features [5,6]. Embedded methods integrate feature selection directly into the learning model, performing selection as part of the model's optimization process [7]. Among these approaches, embedded methods typically achieve better feature selection while maintaining lower computational costs [8]. Recent multi-label feature selection methods utilize insights into label correlation in label manifold structures to preserve consistency between the label space and predicted labels [9,10]. These correlations capture the structural relationships among labels, enabling models to exploit inter-label information for more accurate and coherent prediction. For instance, consider the labels "rain" and "coat"; their co-occurrence increases the likelihood of related labels, so the label "umbrella" is also

* Corresponding author.
E-mail addresses: mohammad.faraji@uok.ac.ir (M. Faraji), seyedamjad.seyedi@umons.ac.be (A. Seyedi), f.akhlaghian@uok.ac.ir (F. Akhlaghian Tab).

https://doi.org/10.1016/j.knosys.2026.115825
Received 29 October 2025; Received in revised form 22 February 2026; Accepted 20 March 2026
Available online 24 March 2026

likely to appear, reflecting meaningful inter-label dependencies. However, the presence of noise can significantly disrupt the learning process, leading to incomplete pattern recognition and reduced prediction accuracy. Moreover, relying on fixed label correlation that includes noisy labels may intensify the issue further by leading to the selection of irrelevant features. Noise can manifest in various forms, such as mislabeling or the inclusion of irrelevant labels. Consider this scenario: if the label "coat" is mistakenly changed to "boat" due to noise, the correlation shifts, leading to the expectation of "river" instead of "umbrella". Such noise often arises from human annotation errors, ambiguous contexts, or algorithmic shortcuts designed to reduce labeling costs [11,12].

In multi-label feature selection, the relationship between features and labels is critical. Embedded regression models are highly effective in capturing this relationship, making them a state-of-the-art approach for multi-label feature selection [13,14]. Their standard least squares loss functions, which utilize the Frobenius norm (i.e., the L_2-norm loss function), are particularly suitable for handling data with Gaussian noises (i.e., zero-mean normal distribution noises). However, in real-world datasets, this assumption often fails, as data typically contain various types of noise that disregard this assumption. To reduce the sensitivity of learning models to noise, recent research in this domain has employed robust norms to improve stability and mitigate the impact of noisy labels [9,10,15]. These approaches use the robust $L_{2,1}$-norm loss function, rather than the Frobenius norm, to measure reconstruction errors. Although the $L_{2,1}$-norm has been theoretically proven to be optimal for handling Laplacian noise, it remains susceptible to extremely large noise values [16]. This limitation arises because the norm treats all errors equally, without distinguishing between small errors and very large ones.

In addition to the negative effects of noisy information, redundancy among features can also adversely affect the learning process. Since the classifier's learning depends directly on the feature values, more features should theoretically lead to better decisions. However, in practice, especially when training data is limited, an excessive number of features can slow down learning and cause overfitting, especially if some features are irrelevant or repeat the same information. Feature relevance is classified into three distinct categories: strongly relevant, weakly relevant, and irrelevant [17]. A strongly relevant feature is indispensable for constructing an optimal subset, as it provides unique and essential information that directly influences the predictive outcome. In contrast, a weakly relevant feature may not always be necessary, but it can still help improve performance when combined with certain other features [18]. Redundant features fall within the scope of weak relevance; although they carry useful information related to the target, this information is already captured by other features, making them unnecessary. Irrelevant features, on the other hand, offer no informative value with respect to the target variable and contribute nothing to the predictive process. An ideal subset should include strongly relevant features, along with some unique weakly relevant ones, and exclude the irrelevant ones [19,20]. Many sparse learning-based feature selection methods use the $L_{2,1}$-norm to enforce sparsity in the feature weight matrix [21,22]. While this norm effectively removes irrelevant features, it does not differentiate between similar features [23]. As a result, redundant features may still be selected during the learning process, which can negatively impact classification performance. Therefore, addressing redundancy among relevant features is essential.

Sparsity in the feature weight matrix plays a vital role in selecting meaningful features. Researchers have explored various norms to induce sparsity, aiming to balance computational efficiency and feature selection quality. Among these, one of the fundamental methods, RALM-FS [24], used the L_0-norm which counts the number of non-zero elements in a matrix to enforce sparsity. However, optimization involving the L_0-norm is computationally challenging due to its combinatorial nature. As a result, the L_1-norm has been proposed and widely adopted in the machine learning community as a convex relaxation of the L_0-norm, offering a more tractable solution. The L_1-norm promotes sparsity by shrinking smaller elements toward zero; however, it can sometimes be an inaccurate approximation of the L_0-norm, particularly when the input matrix or vector contains large entries. Recent multi-label feature selection methods [13,21,25] have employed the $L_{2,1}$-norm to induce column-wise sparsity, which is beneficial for estimating feature weights and selecting relevant features. Unlike the L_1-norm, which treats each element of the matrix independently, the $L_{2,1}$-norm sums the L_2-norm of each row (or column) equally, preserving the structural relationships within each entity. Since the $L_{2,1}$-norm is associated with the L_1-norm, both are affected by large values, potentially leading to biased sparsity toward high-value feature weights and making it challenging to achieve true sparsity [26]. Additionally, both norms fail to distinguish between similar features, which may result in retaining both features and causing redundancy in the selected feature set [14].

In summary, despite notable progress in multi-label feature selection, existing methods remain constrained by three core issues: (1) limited robustness of commonly used loss functions to large or non-Gaussian noise, (2) static modeling of label correlations that fails to adapt to noisy or incomplete labels, and (3) sparsity mechanisms that inadequately account for redundancy among relevant features. These limitations prevent current approaches from simultaneously achieving robustness, adaptivity, and effective relevance-redundancy trade-offs. This observation motivates the need for a unified feature selection framework that can robustly handle noisy labels, dynamically model label dependencies, and explicitly optimize feature relevance and redundancy. To directly address the above limitations, we propose Robust log-based multi-label feature selection with dynamic label correlation and relevance-redundancy optimization (RLBMLFS). The proposed framework introduces three tightly coupled components that correspond to the identified gaps. First, to overcome the sensitivity of existing loss functions to large or non-Gaussian noise, we design a robust logarithmic element-wise loss (Hx) that suppresses the influence of extreme reconstruction errors more effectively than sample-wise $L_{2,1}$-norm losses. Second, to mitigate error propagation caused by static label dependencies, we learn dynamic label correlations directly from noisy labels and integrate them into the label manifold, enabling adaptive and noise-resilient modeling of label relationships. Third, to jointly handle feature relevance and redundancy, we incorporate a logarithmic redundancy-aware penalty together with an $L_{2,\log}$ pseudo-norm sparsity term, which provides a closer approximation to L_0-norm sparsity than $L_1/L_{2,1}$-norms while explicitly discouraging the selection of highly correlated features. Collectively, these components form a unified algorithmic framework that simultaneously enhances robustness, adaptivity, and relevance-redundancy discrimination in multi-label feature selection. The key contributions of this paper are highlighted as follows:

- We utilize a highly robust loss function to reliably evaluate reconstruction errors while reducing sensitivity to noisy data. This loss function is able to suppress high reconstruction errors, significantly reducing their influence on the overall loss and effectively mitigating the impact of noisy labels.
- We extract dynamic label correlations instead of fixed label correlation to enhance the reliability of label correlation. By leveraging high-quality correlations within the label manifold, the model achieves more accurate label predictions and improves the effectiveness of the feature selection process.
- To mitigate the adverse effects of feature redundancy on the learning process, we incorporate a logarithmic penalty term that systematically evaluates each feature, suppresses the selection of similar features, and thereby selects a more informative and non-redundant feature set.
- We utilize a logarithmic sparsity term, the $L_{2,\log}$ pseudo-norm, on the feature matrix to enforce more appropriate sparsity and effectively select informative features.

Table 1
Summary of the all notations.

Symbol	Definition
a	A scalar.
$\boldsymbol{b}$	A vector.
$\boldsymbol{X} \in \mathbb{R}^{n\times d}$	Feature matrix.
$\boldsymbol{Y} \in \mathbb{R}^{n\times l}$	Label matrix.
Y_{ij}	The (i,j)th element of $\boldsymbol{Y}$.
$\boldsymbol{W} \in \mathbb{R}^{d\times l}$	Feature weight matrix.
E_{ij}	Reconstruction error at position (i,j).
$\langle \boldsymbol{W}^{(i)}, \boldsymbol{W}^{(j)} \rangle$	Inner product between feature vectors i and j.
$\boldsymbol{C}^{(0)}$	Initial label correlation matrix (Cosine Similarity).
$\boldsymbol{C} \in \mathbb{R}^{l\times l}$	Label correlation matrix.
$\boldsymbol{L}_C$	Laplacian matrix.
$\boldsymbol{A}$	Diagonal matrix.
Trace$(\cdot)$	Trace of a matrix.
$(\cdot)_{H_x}$	Simplification of $\sum_{i,j} \log(1+\lvert.\rvert)$.
$\mathscr{F}(\cdot)$	An increasing function.
$\lvert\cdot\rvert$	Absolute value operator.

- To effectively optimize the unified cost function, we propose an Iterative Reweighted Algorithm that incorporates efficient update rules for robust and reliable performance.
- The effectiveness of the RLBMLFS is evaluated using 17 multi-label datasets and compared against 13 state-of-the-art multi-label feature selection methods. The experimental results highlight the framework's competitive performance across five evaluation metrics.

The rest of the paper is organized as follows: Section 2 covers fundamental concepts and related work. Section 3 provides a detailed explanation of the proposed model and its optimization equations. Section 4 presents an analysis of the experimental results. Section 5 summarizes the key findings of the study and discusses potential areas for future research and development.

2. Preliminaries

In this paper, bold uppercase letters are used to denote matrices, such as $\boldsymbol{A}$. When $\boldsymbol{A} \in \mathbb{R}^{n\times m}$, n represents the number of rows, and m represents the number of columns. Specifically, for the data matrix $\boldsymbol{X} \in \mathbb{R}^{n\times d}$, n denotes the number of instances, and each instance is described by an d-dimensional feature space. Similarly, $\boldsymbol{Y} \in \mathbb{R}^{n\times l}$ represents the same n instances with an l-dimensional label space. If the ith sample has the j-label, then $Y_{ij} = 1$; otherwise, $Y_{ij} = 0$. We denote $\boldsymbol{a}_i$ and $\boldsymbol{a}^{(j)}$ as the ith column and the jth row of $\boldsymbol{A}$, respectively. The transpose and trace of $\boldsymbol{A}$ denoted as $\boldsymbol{A}^\top$ and Tr$(\boldsymbol{A})$, respectively. We denote the Frobenius norm and $L_{2,1}$-norm of a matrix $\boldsymbol{A}$ by $\|\boldsymbol{A}\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{d} A_{ij}^2}$ and $\|\boldsymbol{A}\|_{2,1} = \sum_{i=1}^{n}\sqrt{\sum_{j=1}^{d} A_{ij}^2}$, respectively, where A_{ij} represents the (i,j)-th entry in matrix $\boldsymbol{A}$. To better demonstrate the notation in this paper, we summarize all notations in Table 1.

2.1. Related work

With the growing proliferation of multi-label datasets, research in multi-label feature selection has progressed significantly. Current state-of-the-art methods can generally be categorized into information-theoretic and embedded-based techniques. Information-theoretic approaches focus on maximizing mutual information, while embedded-based methods integrate feature selection directly into the learning process. Despite their effectiveness, both approaches are often vulnerable to the disruptive effects of noisy labels, which can significantly undermine their performance [13]. In embedded-based methods, noisy labels can interfere with the learning process, resulting in a suboptimal feature weight matrix. In particular, some methods rely on label correlation to enhance the feature selection process, but in the presence of noise, the label matrix becomes contaminated, distorting the correlation. Consequently, the model may make inaccurate and unreliable feature selections.

In this section, we provide a concise overview of multi-label feature selection methods, focusing on information-theoretic, embedded, and robust embedded approaches. One of the well-known information-theoretic methods is SCLS [27], which includes a feature relevance term and offers scalable relevance evaluation.

$$J(\boldsymbol{f}_k) = \sum_{\boldsymbol{l}_i \in L} I(\boldsymbol{f}_k;\boldsymbol{l}_i) - \sum_{\boldsymbol{f}_j \in S} \frac{I(\boldsymbol{f}_k;\boldsymbol{f}_j)}{H(\boldsymbol{f}_k)} \sum_{\boldsymbol{l}_i \in L} I(\boldsymbol{f}_k;\boldsymbol{l}_i), \tag{1}$$

where $H(\boldsymbol{f}_k)$ denotes the entropy of the kth feature. Another information-theoretic method is the Label Redundancy-based Feature Selection (LRFS) approach, in which labels are divided into independent and dependent groups [28].

Information-theoretic-based methods often fail to estimate complex, high-order relationships between features and labels. The effectiveness of these approaches depends on how well they account for the individual importance of each feature. In contrast, embedded-based multi-label feature selection approaches utilize label correlation along with feature-label relationships to select a compact feature subset. In recent years, various sparse embedded-based feature selection techniques have been proposed [5,24,25]. Among these methods, Nie et al. [24] introduced Robust Feature Selection (RFS), which utilizes joint $L_{2,1}$-norm minimization to improve feature selection performance. The method is formulated as follows:

$$\min_{\boldsymbol{W}} \|\boldsymbol{X}\boldsymbol{W} - \boldsymbol{Y}\|_{2,1} + \gamma\|\boldsymbol{W}\|_{2,1}, \tag{2}$$

where $\boldsymbol{W} \in \mathbb{R}^{d\times c}$ is a coefficient matrix, and the most discriminating features are chosen using more effective sparseness term $\|\boldsymbol{W}\|_{2,1}$. Although RFS was initially designed for multi-class feature selection, its framework also performs well in multi-label cases. Due to this flexibility, it has been widely used in various multi-label feature selection methods, such as [29], which extendeds the RFS model to address Missing Label Multi-Label Feature Selection (MLMLFS). This method incorporated a robust linear regression approach combined with graph regularization, under the assumption that similar instances share similar labels. Furthermore, a subset of discriminative features was selected by applying an $L_{2,p}$ norm constraint, where $0 < p \leq 1$. Since learning multi-label regression models on a binary label matrix is inherently challenging, recent research has explored alternative representations of the label matrix. These approaches generally fall into two categories: utilizing a latent label matrix or a pseudo-label matrix as the regression goal. One well-known latent-based method is Multi-label Informed Feature Selection (MIFS) [25], which leverages implicit label correlations to identify the most relevant features. Additionally, MIFS considers a low-dimensional reduced label matrix to mitigate the exponential increase in the number of features and labels. The formulation of MIFS is expressed as follows:

$$\min_{\boldsymbol{W},\boldsymbol{V},\boldsymbol{B}} \|\boldsymbol{X}\boldsymbol{W} - \boldsymbol{V}\|_F^2 + \alpha\|\boldsymbol{Y} - \boldsymbol{V}\boldsymbol{B}\|_F^2 + \beta\mathrm{Tr}(\boldsymbol{V}^\top\boldsymbol{L}\boldsymbol{V}) + \gamma\|\boldsymbol{W}\|_{2,1}, \tag{3}$$

where $\boldsymbol{X}$ and $\boldsymbol{W}$ are feature matrix and coefficient matrix of features, respectively. $\boldsymbol{Y}$ is a label matrix and $\boldsymbol{V}$ is latent matrix, whose coefficient matrix is denoted by $\boldsymbol{B}$. The MIFS method aims to extract shared latent information from both feature and label matrices. This foundational concept has influenced several cutting-edge multi-label feature selection techniques. Specifically, MIFS factorizes the multi-label matrix into two component matrices, each containing entries with mixed signs, which can make interpretation challenging. Shared Common Mode Feature Selection (SCMFS) [3] is another variation of MIFS that similarly captures shared latent structures between the feature and label spaces using an NMF-based model. To enhance the validity of the matrix regression term, SCMFS further incorporates a decoder factorization term on the feature matrix.

$$\min_{\boldsymbol{V},\boldsymbol{B},\boldsymbol{W},\boldsymbol{Q}\geq 0} \|\boldsymbol{X}\boldsymbol{W} - \boldsymbol{V}\|_F^2 + \alpha\|\boldsymbol{X} - \boldsymbol{V}\boldsymbol{Q}\|_F^2 + \beta\|\boldsymbol{Y} - \boldsymbol{V}\boldsymbol{B}\|_F^2 + \gamma\|\boldsymbol{W}\|_{2,1}. \tag{4}$$

More recently, Gao et al. [21] proposed Shared Structure Feature Selection (SSFS), which replaces the regression term (encoder structure)

in MIFS with a factorization term (decoder structure). Like SCMFS, it employs an NMF-based model to optimize the cost function.

$$\min_{\boldsymbol{V},\boldsymbol{Q},\boldsymbol{M}\geq 0}\|\boldsymbol{X}-\boldsymbol{V}\boldsymbol{Q}^{\top}\|_F^2+\alpha\|\boldsymbol{Y}-\boldsymbol{V}\boldsymbol{M}\|_F^2+\beta\mathrm{Tr}(\boldsymbol{V}^{\top}\boldsymbol{L}\boldsymbol{V})+\gamma\|\boldsymbol{Q}\|_{2,1}. \tag{5}$$

He et al. [30] proposed the Multi-label Feature Selection via Similarity Constraints with Non-negative Matrix Factorization (SCNMF). In this method, non-negative matrix factorization is employed to decompose the similarity matrix into a weight matrix, while inner-product constraints are imposed to better preserve similarity information. In addition, a label-similarity regularization term is incorporated to enhance the learning of label weights. The SCNMF optimization problem is formulated as follows:

$$\begin{aligned}&\min_{\boldsymbol{W}}\|\boldsymbol{X}\boldsymbol{W}-\boldsymbol{Y}\|_F^2+\lambda_1\|\boldsymbol{W}\boldsymbol{W}^{\top}-\boldsymbol{H}\|_F^2+\lambda_2\mathrm{Tr}(\boldsymbol{L}\boldsymbol{W}^{\top}\boldsymbol{W})\quad+\lambda_3\|\boldsymbol{W}\|_{2,1},\\ &s.t.\quad \boldsymbol{W}\geq 0.\end{aligned} \tag{6}$$

More recently, Wu and Bai [31] presented a Sparse and Low-redundancy multi-label feature selection method with Constrained Laplacian Rank, abbreviated as SLCLR. This method employs an L_1-norm sparsity to remove irrelevant features and combines L_1-norm and $L_{2,1}$-norm regularization to eliminate redundant features. Moreover, SLCLR introduces a Laplacian rank constraint to construct a dynamic graph, thereby preserving the manifold structure of the feature space and improving feature selection performance. The formulation of SLCLR is formulated as follows,

$$\begin{aligned}&\min_{\boldsymbol{W},\boldsymbol{S},b}\|\boldsymbol{X}\boldsymbol{W}+1_n b-\boldsymbol{Y}\|_F^2+\alpha\|\boldsymbol{S}-\boldsymbol{A}\|_1+\beta\mathrm{Tr}(\boldsymbol{W}^{\top}\boldsymbol{L}_s\boldsymbol{W})+\gamma(\|\boldsymbol{W}\boldsymbol{W}^{\top}\|_1+\|W\|_{2,1}),\\ &s.t.\quad S_{ij}\geq 0,\sum_j S_{ij}=1,\boldsymbol{W}\geq 0.\end{aligned} \tag{7}$$

There are various pseudo-label-based methods in the MLFS literature that make connections between the label matrix and its alternatives in different ways. For instance, Huang and Wu [22] proposed an MLFS method called Manifold Regularization and Dependence Maximization (MRDM), which replaces the label space with a manifold embedding. Additionally, it leverages the Hilbert-Schmidt Independence Criterion (HSIC) as a regularization technique to maximize the dependence between the manifold embedding and the original label matrix.

$$\min_{\boldsymbol{W},\boldsymbol{Z}^{\top}\boldsymbol{Z}=\boldsymbol{I}}\|\boldsymbol{X}\boldsymbol{W}-\boldsymbol{Z}\|_F^2+\alpha\mathrm{Tr}(\boldsymbol{Z}^{\top}\boldsymbol{L}\boldsymbol{Z})-\beta\mathrm{Tr}(\boldsymbol{H}\boldsymbol{Z}\boldsymbol{Z}^{\top}\boldsymbol{H}\boldsymbol{Y}\boldsymbol{Y}^{\top})+\gamma\|\boldsymbol{W}\|_{2,1}. \tag{8}$$

More recently, Faraji et al. [13] proposed the Multi-Label Feature Selection with Global and Local Label Correlation (MLFS-GLOCAL) method. This method learns the shared common structure between the feature and label matrices to capture implicit label correlation. To extract explicit label correlation, it incorporates both global and local label correlation, which enhances the feature selection process and helps identify the most relevant features for each label.

$$\begin{aligned}&\min_{\boldsymbol{V},\boldsymbol{B},\boldsymbol{W}\geq 0}\|\boldsymbol{Y}-\boldsymbol{V}\boldsymbol{B}\|_F^2+\|\boldsymbol{V}-\boldsymbol{X}\boldsymbol{W}\|_F^2+\lambda_1\mathrm{Tr}(\boldsymbol{V}^{\top}\boldsymbol{L}\boldsymbol{V})+\lambda_2[\mathrm{Tr}(\boldsymbol{F}\boldsymbol{P}\boldsymbol{F}^{\top})\\ &+\sum_{m=1}^{g}\frac{n_m}{n}\mathrm{Tr}(\boldsymbol{F}_m\boldsymbol{P}_m\boldsymbol{F}_m^{\top})]+\lambda_3\|\boldsymbol{W}\|_{2,1},\quad \boldsymbol{F}=\boldsymbol{X}\boldsymbol{W}\boldsymbol{B},\boldsymbol{F}_m=\boldsymbol{X}_m\boldsymbol{W}\boldsymbol{B}.\end{aligned} \tag{9}$$

In recent robust multi-label feature selection methods that use the $L_{2,1}$-norm as a robust loss function, the model treats all labels in a sample as noisy when an error is detected, thereby reducing their impact during the training process. For instance, Hu et al. [9] proposed a robust multi-label feature selection method with dual-graph regularization (DRMFS). The DRMFS method uses both feature graph regularization and label graph regularization to preserve the geometric relationships between features and labels in the feature-label space. Additionally, the $L_{2,1}$-norm is applied to the loss function to improve robustness against long-tailed label noise during the feature selection process.

$$\min_{\boldsymbol{W}\geq 0}\|\boldsymbol{X}^{\top}\boldsymbol{W}-\boldsymbol{Y}\|_{2,1}+\alpha\mathrm{Tr}(\boldsymbol{W}^{\top}\boldsymbol{L}^{X}\boldsymbol{W})+\beta\mathrm{Tr}(\boldsymbol{W}\boldsymbol{L}^{Y}\boldsymbol{W}^{\top})+\gamma\|\boldsymbol{W}\|_{2,1}. \tag{10}$$

Li et al. [32] presented a robust and flexible sparse feature selection method called RFSFS. This method leverages global label correlation to exploit the relationships between labels and simultaneously applies both the L_1-norm and L_2-norm to reduce redundancy among selected features.

$$\min_{\boldsymbol{W}\geq 0}\|\boldsymbol{X}\boldsymbol{W}-\boldsymbol{Y}\|_F^2+\alpha\mathrm{Tr}(\boldsymbol{W}\boldsymbol{L}_{\boldsymbol{C}}\boldsymbol{W}^{\top})+\beta\|\boldsymbol{W}\|_1+\gamma(\|\boldsymbol{W}\boldsymbol{W}^{\top}\|_1-\|\boldsymbol{W}\|_F^2). \tag{11}$$

Also, Li et al. [33] proposed a label correlation variation-based method for robust multi-label feature selection. It employs self-expression to construct a self-representation matrix, where an $L_{2,1}$-norm on $\boldsymbol{Z}$ mitigates redundancy among features and reduce the noise information.

$$\min_{\boldsymbol{W},\boldsymbol{Z}\geq 0}\|\boldsymbol{X}\boldsymbol{W}-\boldsymbol{Y}\boldsymbol{Z}\|_F^2+\lambda_1\|\boldsymbol{Y}-\boldsymbol{Y}\boldsymbol{Z}\|_F^2+\lambda_2\mathrm{Tr}((\boldsymbol{Y}\boldsymbol{Z})\boldsymbol{L}(\boldsymbol{Y}\boldsymbol{Z})^{\top})+\lambda_3\|\boldsymbol{Z}\|_{2,1}+\lambda_4\|\boldsymbol{W}\|_1. \tag{12}$$

Recently, a fusion-enhanced multi-label feature selection method with sparse supplementation proposed [23]. In this method to overcome the limitations of the L_1-norm and $L_{2,1}$-norm, utilizes a joint formulation that combines the $L_{2,1}$-norm with a sparse supplementation term as a fusion component.

$$\min_{\boldsymbol{W}\geq 0}\|\boldsymbol{X}\boldsymbol{W}-\boldsymbol{Y}\|_F^2+\alpha\mathrm{Tr}(\boldsymbol{W}\boldsymbol{L}\boldsymbol{W}^{\top})+\beta\|\boldsymbol{W}\|_{2,1}+\gamma(\|\boldsymbol{W}\boldsymbol{W}^{\top}\|_1-\|\boldsymbol{W}\|_F^2). \tag{13}$$

More recently, the GNN-DE [34] method integrates Graph Neural Network (GNN) to capture complex feature-label and label-label dependencies with Differential Evolution (DE) to efficiently search for an optimal feature subset. This hybrid design leverages deep representation learning for relevance modeling while maintaining robust and scalable optimization for multi-label feature selection.

Existing methods provide valuable frameworks for multi-label feature selection but have notable limitations. The $L_{2,1}$-norm robust loss function is widely used to reduce noise, yet it struggles with high noise levels, treating entire label vectors as noisy if any label is affected, which is not an optimal solution. While leveraging label correlation can enhance the feature weight matrix, its effectiveness diminishes in noisy data, potentially leading to misleading results. The $L_{2,1}$-norm promotes sparsity while preserving structural relationships better than the L_1-norm, but it remains sensitive to large values and may struggle to differentiate similar features.

2.2. Iterative reweighted algorithm

The Iterative Reweighted Algorithm (IRA) is a simple yet powerful method widely used for optimizing non-convex models [16]. Instead of directly minimizing a complex non-quadratic objective function, which can be computationally expensive, IRA reformulates the problem as a sum of weighted least squares subproblems. By leveraging efficient linear algebra techniques, sparse matrix computations, and parallel processing, IRA significantly improves computational efficiency. Specifically, in the context of element-wise factorization, the reconstruction error for the (i,j)th element of a matrix, denoted as E_{ij}, represents the difference between its true value and the reconstructed value. The error for element (i,j) is formulated as:

$$E_{ij}=|Y_{ij}-[XW]_{ij}|, \tag{14}$$

where Y_{ij} is the true element in the label matrix and $[XW]_{ij}$ is its reconstruction. The general reconstruction problem can be formulated as the following objective function:

$$\min_{q}\sum_{i=1}^{n}\sum_{j=1}^{l}\mathscr{F}(E_{ij}(\boldsymbol{q})), \tag{15}$$

where $\mathscr{F}(.)$ is an increasing function and $\boldsymbol{q}=[q_1,q_2,\ldots,q_p]^{\top}$ contains p unknown variables, to be computed by solving formula (15). In the element-wise model (14), the vector $\boldsymbol{q}$ represents the entries of matrix

$\boldsymbol{W}$. To achieve the optimal solution, the derivative of Eq. (15) with respect to $\boldsymbol{q}$ is set to zero, as shown below:

$$\sum_{i=1}^{n}\sum_{j=1}^{l}\Omega(E_{ij}(\boldsymbol{q}))\frac{\partial E_{ij}(\boldsymbol{q})}{\partial q_k}=0,\quad k=1,2,\ldots,p, \tag{16}$$

where $\Omega(E_{ij}(\boldsymbol{q}))=\frac{d\mathscr{F}(E_{ij}(\boldsymbol{q}))}{dE_{ij}(\boldsymbol{q})}$ is called influence function. Additionally, the formula (16) can be written as :

$$\sum_{i=1}^{n}\sum_{j=1}^{l}\omega(E_{ij}(\boldsymbol{q}))E_{ij}(\boldsymbol{q})\frac{\partial E_{ij}(\boldsymbol{q})}{\partial q_k}=0,\quad k=1,2,\ldots,p, \tag{17}$$

where $\omega(E_{ij}(\boldsymbol{q}))$ is known as the weight function and is expressed as:

$$\omega(E_{ij}(\boldsymbol{q}))=\frac{\Omega(E(\boldsymbol{q}))}{E(\boldsymbol{q})}=\frac{\mathscr{F}'(E(\boldsymbol{q}))}{E(\boldsymbol{q})}. \tag{18}$$

Therefore, Eq. (17) serves as the solution to the following iterative reweighted problem:

$$\min_{\boldsymbol{q}}\sum_{i=1}^{n}\sum_{j=1}^{l}\omega(E_{ij}(\boldsymbol{q})^{[t-1]})E_{ij}(\boldsymbol{q})^2, \tag{19}$$

where $E_{ij}(\boldsymbol{q})^{[t-1]}$ denotes the reconstruction error from the $(t-1)$th iteration. The process of solving problem (19) at each iteration can be divided into two steps. First, treat the weight $\omega(E_{ij}(\boldsymbol{q})^{[t-1]})$ as a constant, and then find the optimal solution according to the specific form of problem (19). Second, update the weight $\omega(E_{ij}(\boldsymbol{q})^{[t-1]})$ using the current reconstruction error $E_{ij}(\boldsymbol{q})^{[t]}$. In Wang et al. [16], a connection was established between the iterative reweighted algorithm and matrix factorization, independent of the loss function used. Specifically, the loss function and $|Y_{ij}-[XW]_{ij}|$ can be interpreted as the $\mathscr{F}(.)$ function and E_{ij} in formula (15), respectively. As a result, we convert the challenging loss function $\min_{\boldsymbol{W}}\sum_{i=1}^{n}\sum_{j=1}^{l}\mathscr{F}(E(Y_{ij},[XW]_{ij}))$ into the following iterative reweighted form:

$$\min_{\boldsymbol{W}\geq 0}\sum_{i=1}^{n}\sum_{j=1}^{l}Z_{ij}(Y_{ij}-[XW]_{ij})^2, \tag{20}$$

where $Z_{ij}=\omega(E_{ij}^{[t-1]})$ represents a coefficient that is updated in each iteration based on Eq. (17), utilizing the residual value from the preceding iteration.

3. Proposed method

In this section, we introduce the Robust log-based multi-label feature selection with dynamic label correlation and relevance-redundancy optimization in detail. The method's success can be attributed to five primary factors: (1) It employs a powerful and robust loss function Hx, which is more robust against noisy data than the previously utilized functions; (2) Leveraging dynamic label correlation, enables the model to effectively capture inter-label relationships, resulting in a more accurate predicted label matrix; (3) It utilizes a logarithmic penalty to address the feature redundancy problem commonly encountered in $L_{2,1}$-norm-based feature selection methods, leading to improved classification performance; (4) It uses the $L_{2,\log}$ pseudo-norm as a sparsity regularization term on the feature weight matrix $\boldsymbol{W}$ to ensure the selection of relevant features. Lastly, it combines all these elements into a single joint learning problem and utilizes an effective alternating minimization approach for optimization. Fig. 1 illustrates the proposed framework, which consists of three key components: log-based loss, log-based sparsity regularization, and dynamic label correlation.

3.1. Robust loss function

Noisy labels can arise from a variety of sources, including human error, ambiguous data, and limitations in automated labeling systems, all of which degrade the performance of machine learning models. As mentioned earlier, Kong et al.[35] proposed the $L_{2,1}$-norm as a robust alternative loss function to the Frobenius norm. The key difference between these two loss functions lies in how they handle reconstruction errors. The Frobenius norm computes the sum of squared errors for all entries in the matrix, which can amplify the effect of noise. In contrast, the $L_{2,1}$-norm computes the L_2-norm for each row and then applies the L_1-norm across the rows, which helps prevent a few noisy data points from dominating the loss function. As a result, the $L_{2,1}$-norm offers improved resilience to noise. Hence, many multi-label feature selection methods based on robust functions have employed the $L_{2,1}$-norm to decrease the effect of noisy labels [9,24,36]. The $L_{2,1}$-norm robust loss function detects noisy labels in a sample-wise manner. In other words, when a noisy label exists in the label matrix, the loss function considers all the labels in that sample, treats the entire sample as noisy, and reduces its impact on the total loss. But this is not an ideal way, because the other labels in the same sample, which are incorrectly treated as noisy, may actually be useful. By downweighting the entire sample, the model risks losing important information from these valid labels, potentially affecting the overall performance and feature selection accuracy. In this paper, we introduce an element-wise general loss function, Hx, which offers greater robustness and performance compared to state-of-the-art feature selection methods. The Hx loss function is defined as follows:

$$\min_{\boldsymbol{W}\geq 0}\sum_{i=1}^{n}\sum_{j=1}^{l}\log(1+|Y_{ij}-[XW]_{ij}|). \tag{21}$$

The above loss function is defined in an element-wise manner, meaning that it operates individually on each element of the matrix rather than addressing the matrix in a sample-wise manner. This allows for more precise handling of errors, as the function can adapt to the specific values of each element. In the element-wise approach, the model can find the beneficial labels more accurately and by utilizing the logarithm function, large reconstruction errors are effectively compressed, reducing their influence on the optimization process [37–39]. To better illustrate the advantages of the Hx loss function in reducing the impact of noisy labels on reconstruction error, we draw the loss functions of L_1, L_2, and Hx in Fig. 2.

Let $e_{ij}=Y_{ij}-[XW]_{ij}$ denote the reconstruction error of the (i,j)th label entry. For the proposed element-wise loss function $\ell(e_{ij})=\log(1+|e_{ij}|)$, the first-order derivative with respect to e_{ij} is given by

$$\frac{\partial\ell(e_{ij})}{\partial e_{ij}}=\frac{\operatorname{sign}(e_{ij})}{1+|e_{ij}|}.$$

This gradient is bounded and monotonically decreasing as $|e_{ij}|$ increases, which implies that large reconstruction errors contribute progressively less to the optimization process. In contrast, the gradient of the L_2-norm loss grows linearly with $|e_{ij}|$, while that of the L_1-norm remains constant, making both more sensitive to large errors. Consequently, the proposed Hx loss suppresses the influence of outliers while remaining sensitive to small and moderate reconstruction errors. Moreover, due to its element-wise formulation, noisy label entries are downweighted individually rather than suppressing all labels of a sample, thereby preserving informative labels within the same instance. This property enhances robustness against label noise and leads to more stable feature selection performance, as illustrated in Fig. 2.

3.2. Dynamic label correlation

For effective feature selection, it is important to use label correlations to identify interdependencies and relationships. However, fixed label correlations are affected by noise in the original labels, which causes them to fail to reflect true label relationships. Additionally, fixed correlations may overlook subtle dependencies, limiting their ability to accurately represent label relationships. For this reason, we use dynamic label correlation, which highlights these interdependencies between the labels during the learning process. The accurate estimation of the label correlation manifold depends on label manifold regularization, which

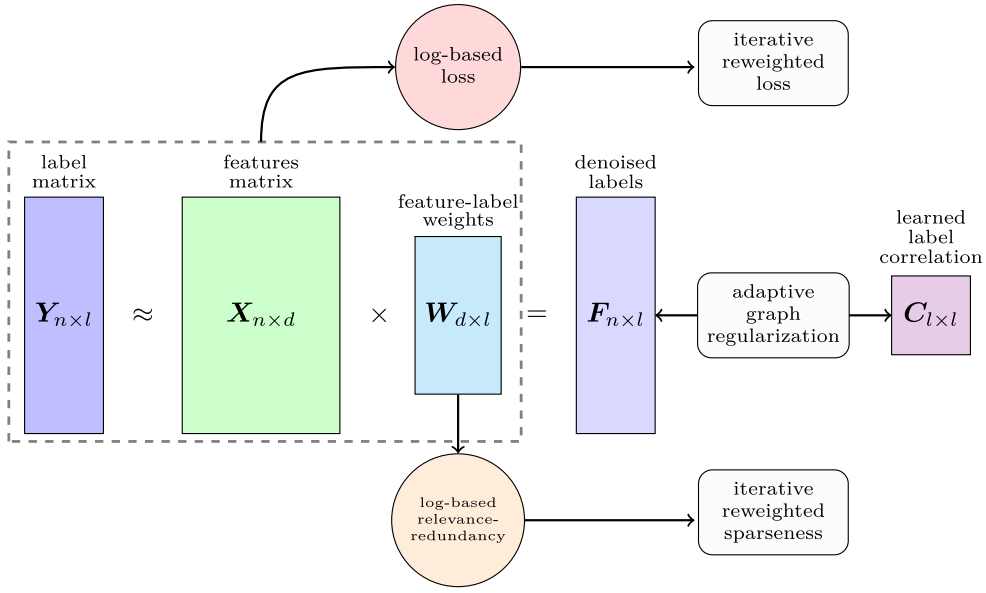


Fig. 1. Schematic illustration of Robust log-based multi-label feature selection with dynamic label correlation and relevance-redundancy optimization.

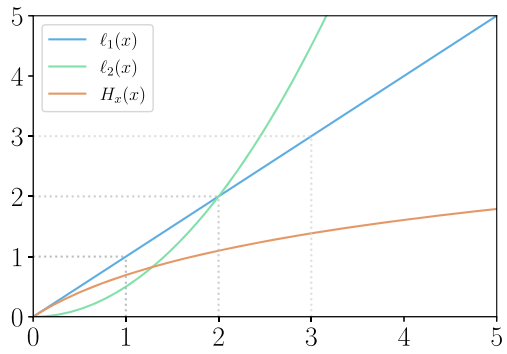


Fig. 2. Comparison curves of L_1-norm, L_2-norm, and Hx loss functions.

should use a high-quality label correlation matrix. In multi-label learning, one rudimentary approach to obtaining the label correlation matrix is by calculating the cosine similarity between the labels [40]. We use this similarity to initialize the label correlation as follows:

$$C_{jk}^{(0)} = \frac{\boldsymbol{y}_j^\top \boldsymbol{y}_k}{||\boldsymbol{y}_j||||\boldsymbol{y}_k||}, \tag{22}$$

where $\boldsymbol{y}_j$ and $\boldsymbol{y}_k$ indicate the jth and kth label vectors corresponding to all instances, and $C_{jk}^{(0)}$ represents the initial label correlation between the jth and kth labels in its fixed form.

In the basic model (29), the label predicted for a sample $\boldsymbol{x}$ is given by $\boldsymbol{xW}$. The feature weight matrix $\boldsymbol{W}$ exhibits a structure similar to the feature matrix $\boldsymbol{X}$ when the label matrix $\boldsymbol{Y}$ is fixed and $\boldsymbol{X}$ changes. Li et al. [23]. Therefore, we use the feature weight matrix $\boldsymbol{W}$ to reflect the relationships between the feature weight matrix and the predicted label matrix. If the jth and kth labels are highly correlated in the label matrix, their corresponding predicted labels, $\boldsymbol{w}_j$ and $\boldsymbol{w}_k$, should be more similar to each other. Therefore, the label manifold regularization is defined as follows:

$$\min_{\boldsymbol{C}\geq 0} \frac{1}{2}\sum_{j=1}^{l}\sum_{k=1}^{l}\|\boldsymbol{w}_j - \boldsymbol{w}_k\|^2 C_{jk} = \mathrm{Tr}(\boldsymbol{WAW}^\top) - \mathrm{Tr}(\boldsymbol{WCW}^\top) = \mathrm{Tr}(\boldsymbol{WL_CW}^\top), \tag{23}$$

where $\boldsymbol{A}$ is a diagonal matrix defined as $A_{jj} = \sum_{k=1}^{l} C_{jk}$, while $\boldsymbol{C}$ is refined and denoised during the learning process, producing a high-quality label correlation matrix that effectively captures intricate interdependencies and underlying relationships between labels. The label Laplacian matrix is given by $\boldsymbol{L_C} = \boldsymbol{A} - \boldsymbol{C}$. By minimizing (23), the term $\|\boldsymbol{w}_j - \boldsymbol{w}_k\|^2$ becomes small for highly correlated labels. The manifold regularizer in (23) is written as $\mathrm{Tr}(\boldsymbol{WL_CW}^\top)$, where $\boldsymbol{W}$ is the classifier output matrix and λ_1 indicates the impact of label correlation on the objective function.

$$\min_{\boldsymbol{W},\boldsymbol{C}\geq 0} \sum_{i=1}^{n}\sum_{j=1}^{l}\log(1 + |Y_{ij} - [XW]_{ij}|) + \lambda_1 \mathrm{Tr}(\boldsymbol{WL_CW}^\top). \tag{24}$$

3.3. Logarithmic penalty

Redundancy among features remains a major problem in multi-label feature selection, as it involves features that convey overlapping or highly correlated information. This redundancy not only increases data dimensionality and adds unnecessary complexity but also diminishes the model's discriminative power. Collectively, these issues hinder the model's ability to generalize effectively and degrade its classification performance [14], making it more challenging to identify and prioritize truly informative patterns. To tackle this challenge, we design a logarithmic penalty term. This term overcomes issues of the L_1-norm discarding features by zeroing weights and the $L_{2,1}$-norm selecting redundant features into optimal subsets. Let define the log-norm (pseudo-norm) as follows :

$$\|\boldsymbol{W}\|_{\log} = \sum_{i=1}^{d}\sum_{j=1}^{l}\log(1 + |W_{ij}|), \tag{25}$$

where the $\log(1 + |W_{ij}|)$ denotes the element-wise log-sum of the matrix $\boldsymbol{W}$. The logarithmic penalty is define as follows:

$$\begin{aligned}\|\boldsymbol{WW}^\top\|_{\log} - \|\boldsymbol{W}\|_F^2 &= \sum_{i,j=1}^{d}\log\left(1 + \left|(\boldsymbol{WW}^\top)^{(i,j)}\right|\right) - \mathrm{Tr}(\boldsymbol{WW}^\top) \\ &= \sum_{i,j=1}^{d}\log\left(1 + \left|\langle \boldsymbol{W}^{(i)}, \boldsymbol{W}^{(j)}\rangle\right|\right) - \sum_{i=1}^{d}|\langle \boldsymbol{W}^{(i)}, \boldsymbol{W}^{(i)}\rangle|.\end{aligned} \tag{26}$$

According to the above formulation, the logarithmic penalty is designed to discourage redundancy between features by penalizing large inner products $\langle \boldsymbol{W}^{(i)}, \boldsymbol{W}^{(j)}\rangle$ between different feature vectors. When two features are highly similar, their inner product is large, which increases the penalty. The diagonal terms $\langle \boldsymbol{W}^{(i)}, \boldsymbol{W}^{(i)}\rangle$ show how strong each feature is by itself. These terms are subtracted so that the penalty focuses

only on how similar different features are to each other. This encourages the model to select more discriminative features and helps avoid learning from redundant information. Minimizing this penalty leads the model to choose features that aren't too similar, resulting in more diverse representations and a more effective learning process, especially in multi-label tasks where feature redundancy is common.

3.4. Log-based sparseness term

As mentioned earlier, the L_0-norm was not an ideal choice for promoting sparsity in the feature weight matrix due to its non-convex nature, which makes optimization difficult. This led to the introduction of the L_1-norm as a convex alternative. However, the L_1-norm does not closely approximate the behavior of the L_0-norm. Recent methods frequently employ the $L_{2,1}$-norm to induce sparsity in the feature matrix [3,13,21], different from L_1 that treats all entries independently, $L_{2,1}$-norm measures the input matrix in a sample-wise manner, which allows it to preserve spatial information of samples [41]. The calculation of the $L_{2,1}$-norm is closely related to the L_1-norm, as it sums the L_2-norms of each row with equal weights. Thus, the $L_{2,1}$-norm may have similar issues to L_1-norm in resulting feature-wise sparse property. In this paper, to address these issues, we utilize the $L_{2,\log}$ pseudo-norm in our approach, which is more effective in inducing sparsity. The $L_{2,\log}$ pseudo-norm has been shown to provide more accurate sparsity by addressing some of the limitations of traditional norms like the L_1-norm and $L_{2,1}$-norm [26]. In the context of feature selection for multi-label learning, applying the $L_{2,\log}$ pseudo-norm to the feature weight matrix allows us to identify the most informative features and ensure that the most relevant features are selected, thereby improving the overall performance of the learning model. Below is the formulation of the $L_{2,\log}$ pseudo-norm:

$$\|\boldsymbol{W}\|_{2,\log} = \sum_{h=1}^{m} \log(1 + \|\boldsymbol{w}^{(h)}\|). \tag{27}$$

This pseudo-norm offers distinct advantages over both the L_0 and L_1 norms across different value ranges. For small values, it provides a smoother transition to zero than L_0's abrupt cutoff, while being more aggressive at eliminating noise than L_1, effectively bridging the gap between them. At moderate values, it creates a critical thresholding effect absent in L_1, forcing coefficients to make a "decision" to either become significant or disappear entirely, while avoiding L_0's computational intractability. For large values, it behaves similarly to L_0 by remaining nearly constant regardless of magnitude increases, unlike L_1, which continues to heavily penalize large coefficients proportionally. This graduated response across value ranges enables the pseudo-norm to recover sparse solutions more effectively than L_1 while remaining computationally feasible, unlike L_0.

As we know, norms impose varying degrees of sparsity on a model's solution. To better understand how the L_0, L_1, and Log norms influence sparsity, we examined their unit ball representations, shown in Fig. 3. These unit balls visually illustrate the feasible regions for each norm, aiding in the comparison of how they promote sparsity and influence feature selection in identifying relevant features.

The L_0-norm creates a "plus" shape, while the L_1-norm forms a "diamond" shape. The L_0-norm promotes sparsity by forcing many coefficients to be zero. However, due to its non-convex nature, it is difficult to optimize directly. In contrast, the L_1-norm offers a more practical solution, balancing sparsity with smoothness. As the edges of the unit ball become sharper, the sparsity of the solution increases. Therefore, to achieve a sparser solution, the unit ball's edges should become sharper. The Log pseudo-norm further refines this by penalizing coefficients more gradually, resulting in even sparser solutions compared to the L_1-norm. As a result, the Log pseudo-norm makes it easier to identify the most relevant features. Finally, we construct the following objective function

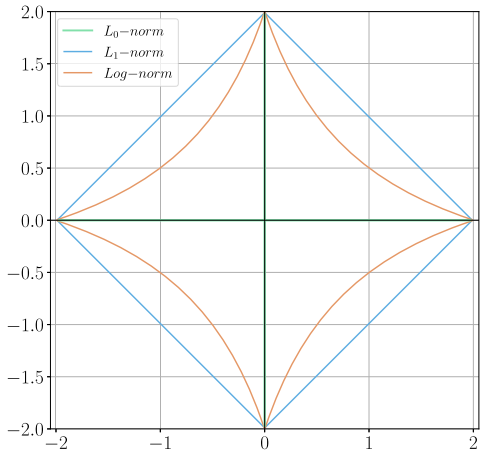


Fig. 3. Comparison of unit balls for sparsity-inducing norms: L_0-norm, L_1-norm, and Log pseudo-norm.

to capitalize on the sparsity-inducing benefits of the $L_{2,\log}$ pseudo-norm:

$$\min_{\boldsymbol{W},\boldsymbol{C}\geq 0} \sum_{i=1}^{n}\sum_{j=1}^{l} \log(1 + |Y_{ij} - [XW]_{ij}|) + \lambda_1 \mathrm{Tr}(\boldsymbol{W}\boldsymbol{L}_{\boldsymbol{C}}\boldsymbol{W}^{\top}) + \lambda_2(\|\boldsymbol{W}\boldsymbol{W}^{\top}\|_{\log} - \|\boldsymbol{W}\|_F^2) + \lambda_3\|\boldsymbol{W}\|_{2,\log}, \tag{28}$$

where the sparsity of $\boldsymbol{W}$ across rows is ensured by using the $L_{2,\log}$ pseudo-norm. The sparsity term is controlled using the λ_2 parameter.

3.5. Optimization

RLBMLFS integrates reliable label correlation and log-based mechanisms including the Hx loss function, a redundancy penalty, and a sparsity-inducing regularization term. These components work together within a unified framework to identify and select the most discriminative features from multi-label data. Before optimizing Eq. (28), several challenges should be addressed. One major issue is that solving the logarithmic function is more difficult than solving the least square function. Additionally, since the objective function involves multiple variables, the optimization problem becomes non-convex, making it difficult to find the global minimum. To address the first issue, inspired by subsection (2.2), we reformulate the function in Eq. (21) as an iterative reweighted optimization problem, effectively mitigating the difficulty of solving the logarithmic function. The reformulation is as follows:

$$\min_{\boldsymbol{W}\geq 0} \sum_{i=1}^{n}\sum_{j=1}^{l} Z_{ij}\left(Y_{ij} - [XW]_{ij}\right)^2, \tag{29}$$

where $Z_{ij} = \frac{1}{E_{ij}(1+E_{ij})+\epsilon}$ is the weight assigned to the (i,j)th element of the label matrix, reflecting the influence of that element's label in the loss function, and $E_{ij} = |Y_{ij} - [XW]_{ij}|$ denotes the reconstruction error. The weight Z_{ij} adjusts the influence of each reconstruction error E_{ij} on the total loss. Specifically, it assigns larger weights to beneficial labels, which have smaller reconstruction errors, and smaller weights to noisy labels, which have larger reconstruction errors. This way, the loss function places more emphasis on the labels that are more accurately predicted and less emphasis on labels that are poorly predicted. This ensures that if the reconstruction error is large, its influence on the overall loss is minimized. Conversely, smaller reconstruction errors are given greater influence, allowing them to have a larger impact on the total loss. The Hx loss function in Eq. (29) can be rewritten as follows:

$$\min_{\boldsymbol{W}\geq 0} \|\boldsymbol{Z}\circ(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})\|_F^2, \tag{30}$$

where $\boldsymbol{Z}\circ(\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})$ is the Hadamard product of $\boldsymbol{Z}$ and $\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W}$.

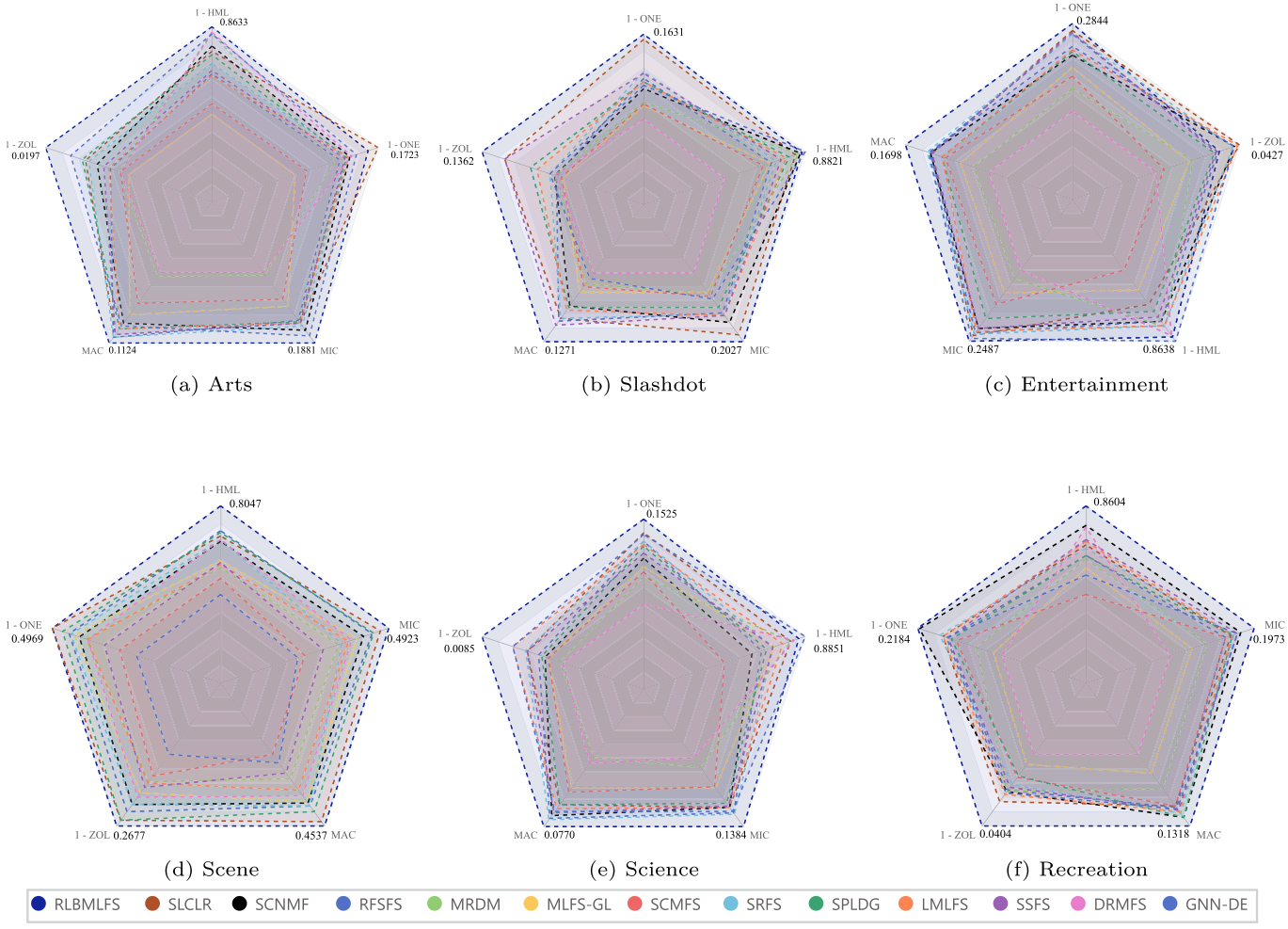


Fig. 4. Comprehensive performance comparison of the proposed method across six multi-label datasets using five evaluation metrics: MIC (Micro-F1), MAC (Macro-F1), Hamming Loss, Zero-One Loss, and One-Error.

The function (28) includes the $L_{2,\log}$ pseudo-norm regularization term, which incorporates a logarithmic component. Similar to Eq. (29), we employ the IRA method (Section 2.2) on the pseudo-norm regularization term, where it reformulated into a weighted version as follows,

$$\|\boldsymbol{W}\|_{2,\log} = \sum_{h=1}^{m} \log(1 + \|\boldsymbol{w}^{(h)}\|) \equiv \sum_{h=1}^{m} d_h \|\boldsymbol{w}^{(h)}\| = \mathrm{Tr}(\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{W}),$$

where $\|\boldsymbol{W}\|_{2,\log}$ is relaxed as $\mathrm{Tr}(\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{W})$, with $\boldsymbol{D}$ being a diagonal matrix [42]. Each diagonal element of $\boldsymbol{D}$ is given by:

$$D_{hh} = \frac{1}{\|\boldsymbol{w}^{(h)}\|(1 + \|\boldsymbol{w}^{(h)}\|) + \epsilon},$$

where $\boldsymbol{w}^{(h)}$ represents the hth row of $\boldsymbol{W}$. As a result, the objective function (28) can be rewritten as follows:

$$\min_{\boldsymbol{W},\boldsymbol{C}\geq 0} \|\boldsymbol{Z}\circ(\boldsymbol{Y} - [\boldsymbol{X}\boldsymbol{W}])\|_F^2 + \lambda_1 \mathrm{Tr}(\boldsymbol{W}\boldsymbol{L}_{\boldsymbol{C}}\boldsymbol{W}^\top) + \lambda_2(\|\boldsymbol{W}\boldsymbol{W}^\top\|_{\log} - \|\boldsymbol{W}\|_F^2) + \lambda_3 \mathrm{Tr}(\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{W}). \tag{31}$$

We use the Lagrangian method to enforce nonnegativity constraint through the Lagrangian multipliers $\boldsymbol{\Omega} \in \mathbb{R}^{d\times l}$ and $\boldsymbol{\Psi} \in \mathbb{R}^{l\times l}$, ensuring that $\boldsymbol{W}$ and $\boldsymbol{C}$ remain restricted. Consequently, Eq. (31) can be reformulated as follows:

$$\Pi = \min_{\boldsymbol{W},\boldsymbol{C}} \|\boldsymbol{Z}\circ(\boldsymbol{Y} - [\boldsymbol{X}\boldsymbol{W}])\|_F^2 + \lambda_1 \mathrm{Tr}(\boldsymbol{W}(\boldsymbol{A} - \boldsymbol{C})\boldsymbol{W}^\top) + \lambda_2(\|\boldsymbol{W}\boldsymbol{W}^\top\|_{\log} - \|\boldsymbol{W}\|_F^2) + \lambda_3 \mathrm{Tr}(\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{W}) - \mathrm{Tr}(\boldsymbol{\Omega}\boldsymbol{W}^\top) - \mathrm{Tr}(\boldsymbol{\Psi}\boldsymbol{C}^\top), \tag{32}$$

where Π represents the objective function. However, since the objective function remains non-convex due to the presence of multiple variables, additional strategies are needed to ensure convergence to a suitable solution. To address this, we fix one variable and update the remaining variables, making the objective function convex in each iteration.

- **Updating feature-label relevancy matrix $\boldsymbol{W}$:**

When updating the feature-label relevancy matrix $\boldsymbol{W}$, we fix $\boldsymbol{C}$ and the remaining variables, and optimize the objective with respect to $\boldsymbol{W}$ only. Under this setting, the Lagrangian in Eq. (32) can be rewritten as the following function of $\boldsymbol{W}$:

$$\Pi(\boldsymbol{W},\boldsymbol{\Omega}) = \mathrm{Tr}\big(\boldsymbol{Z}\circ(\boldsymbol{Y} - [\boldsymbol{X}\boldsymbol{W}])\boldsymbol{Z}^\top\circ(\boldsymbol{Y}^\top - [\boldsymbol{W}^\top\boldsymbol{X}^\top])\big) + \lambda_1 \mathrm{Tr}(\boldsymbol{W}(\boldsymbol{A} - \boldsymbol{C})\boldsymbol{W}^\top) + \lambda_2 \sum_{i,j} \log(1 + (\boldsymbol{W}\boldsymbol{W}^\top)_{ij}) - \lambda_2 \mathrm{Tr}(\boldsymbol{W}\boldsymbol{W}^\top) + \lambda_3 \mathrm{Tr}(\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{W}) - \mathrm{Tr}(\boldsymbol{\Omega}\boldsymbol{W}^\top) \tag{33}$$

The partial derivatives of Eq. (32) with respect to $\boldsymbol{W}$ are represented as follows:

$$\frac{\partial \Pi}{\partial \boldsymbol{W}} = -\boldsymbol{X}^\top(\boldsymbol{Z}\circ\boldsymbol{Y}) + \boldsymbol{X}^\top(\boldsymbol{Z}\circ[\boldsymbol{X}\boldsymbol{W}]) + 2\lambda_1(\boldsymbol{W}(\boldsymbol{A} - \boldsymbol{C})) + 2\lambda_2 \frac{1}{1 + \boldsymbol{W}\boldsymbol{W}^\top}\boldsymbol{W} - 2\lambda_2\boldsymbol{W} + 2\lambda_3\boldsymbol{D}\boldsymbol{W} - \boldsymbol{\Omega}. \tag{34}$$

By setting the partial derivatives in Eq. (34) to zero, We can find the critical points of the function. According to the Karush-Kuhn-Tucker (KKT) conditions [43–45], we set $\boldsymbol{W}_{ij}\circ\boldsymbol{\Omega}_{ij} = 0$, which defines a fixed-point equation that the solution must satisfy at convergence. By solving

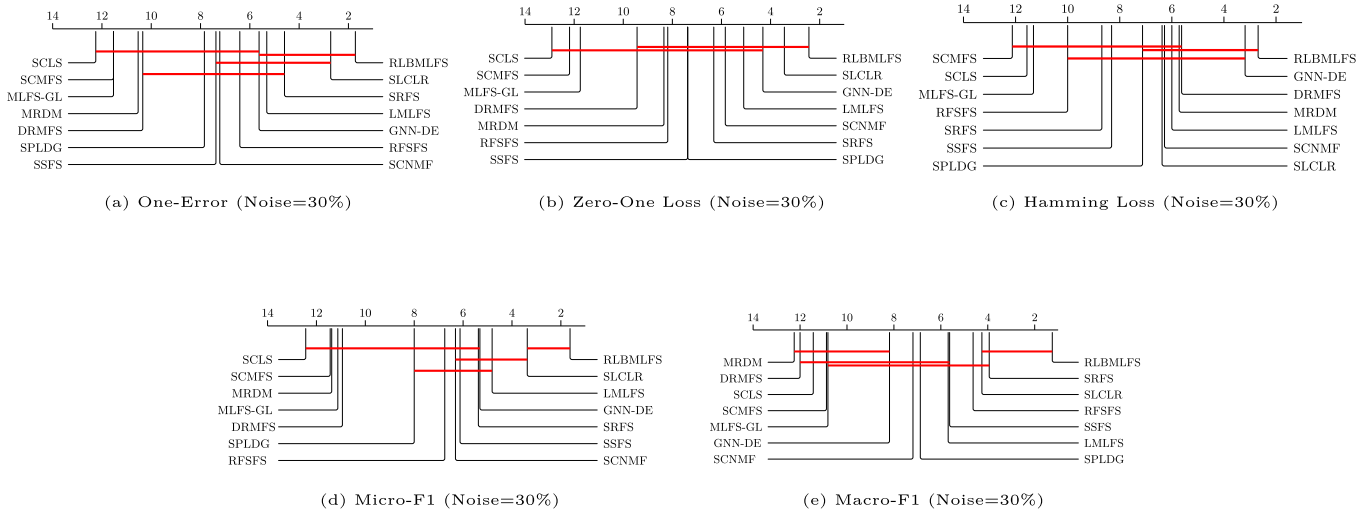


Fig. 5. Comparison of our method against other feature selection methods with the Bonferroni-Dunn test.

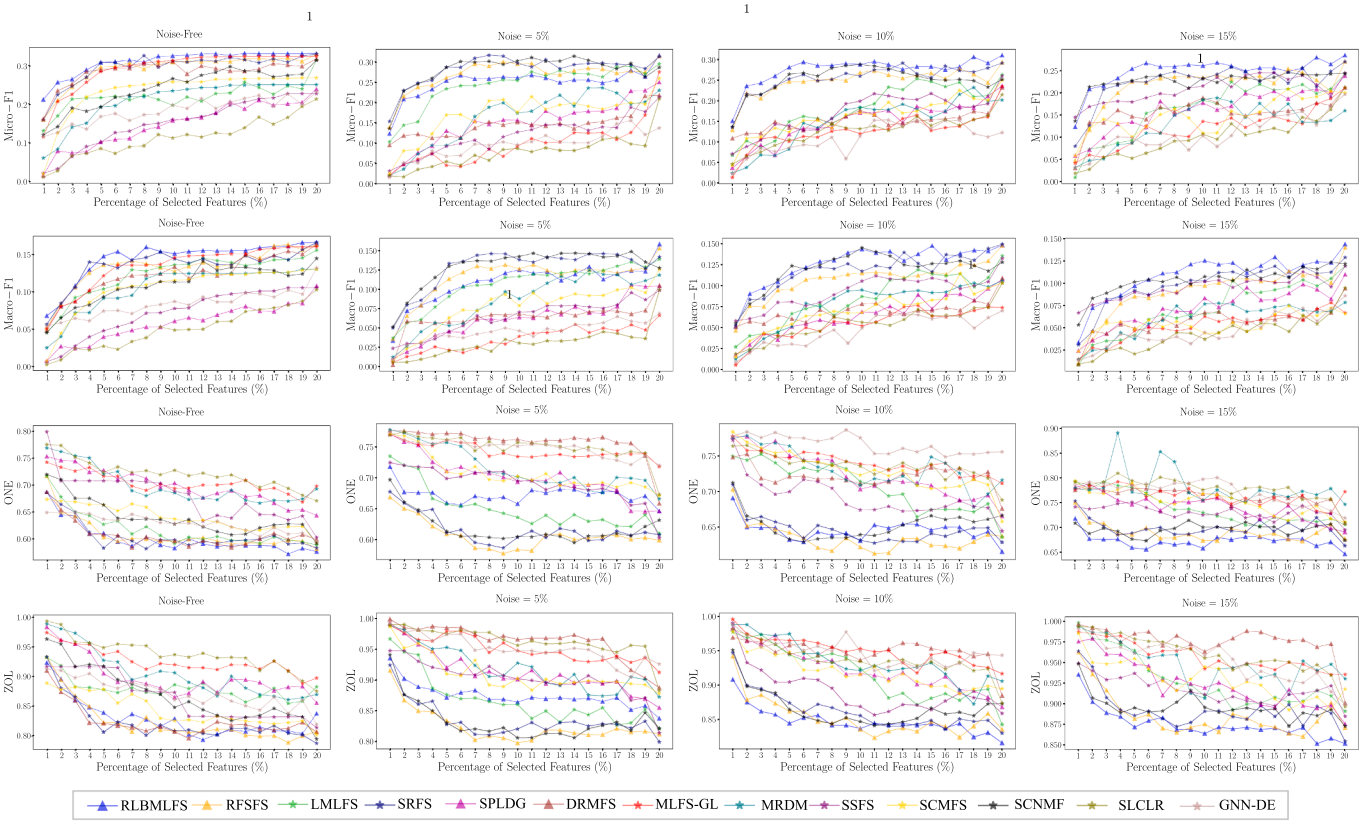


Fig. 6. Robustness analysis on Arts dataset in terms of Micro-F1, Macro-F1, One-Error, and Zero-One Loss.

this equation, we derive the following update rule for matrix $\boldsymbol{W}$:

$$\boldsymbol{W} \leftarrow \boldsymbol{W} \circ \frac{\boldsymbol{X}^\top(\boldsymbol{Z}\circ\boldsymbol{Y}) + \lambda_1 \boldsymbol{W}\boldsymbol{C} + \lambda_2 \boldsymbol{W}}{\boldsymbol{X}^\top(\boldsymbol{Z}\circ[\boldsymbol{X}\boldsymbol{W}]) + \lambda_1 \boldsymbol{W}\boldsymbol{A} + \lambda_2 \frac{1}{1+\boldsymbol{W}\boldsymbol{W}^\top}\boldsymbol{W} + \lambda_3 \boldsymbol{D}\boldsymbol{W}}. \tag{35}$$

- **Updating correlation matrix $\boldsymbol{C}$:**

Based on Eq. (23), after omitting terms independent of C_{jk}, the loss function simplifies to:

$$\begin{aligned}\Pi_C &= \mathrm{Tr}(\boldsymbol{W}\boldsymbol{A}\boldsymbol{W}^\top) - \mathrm{Tr}(\boldsymbol{W}\boldsymbol{C}\boldsymbol{W}^\top) - \mathrm{Tr}(\boldsymbol{\Psi}\boldsymbol{C}^\top) \\ &= \sum_{j=1}^{l} \boldsymbol{w}_j^\top \boldsymbol{w}_j \sum_{k=1}^{l} C_{jk} - \sum_{j=1}^{l}\sum_{k=1}^{l} \boldsymbol{w}_j^\top \boldsymbol{w}_k C_{jk} - \sum_{j=1}^{l}\sum_{k=1}^{l} \Psi_{jk} C_{jk}.\end{aligned} \tag{36}$$

By taking the partial derivatives with respect to C_{jk} from function (36), we obtain the following formula:

$$\frac{\partial \Pi_C}{\partial C_{jk}} = \boldsymbol{w}_j^\top \boldsymbol{w}_j - \boldsymbol{w}_j^\top \boldsymbol{w}_k - \Psi_{jk}. \tag{37}$$

Similar to (35) and based on the above formula, we derive the following update rules for $\boldsymbol{C}$:

$$\boldsymbol{C} \leftarrow \boldsymbol{C} \circ \frac{\boldsymbol{W}^\top \boldsymbol{W}}{\mathrm{diag}(\boldsymbol{W}^\top \boldsymbol{W})\boldsymbol{e}^\top}. \tag{38}$$

where $\boldsymbol{e}$ denotes the all-ones vector. RLBMLFS ranks all features based on $\|\boldsymbol{w}^{(h)}\|$ where $h \in \{1, \ldots, m\}$ in descending order, allowing the selec-

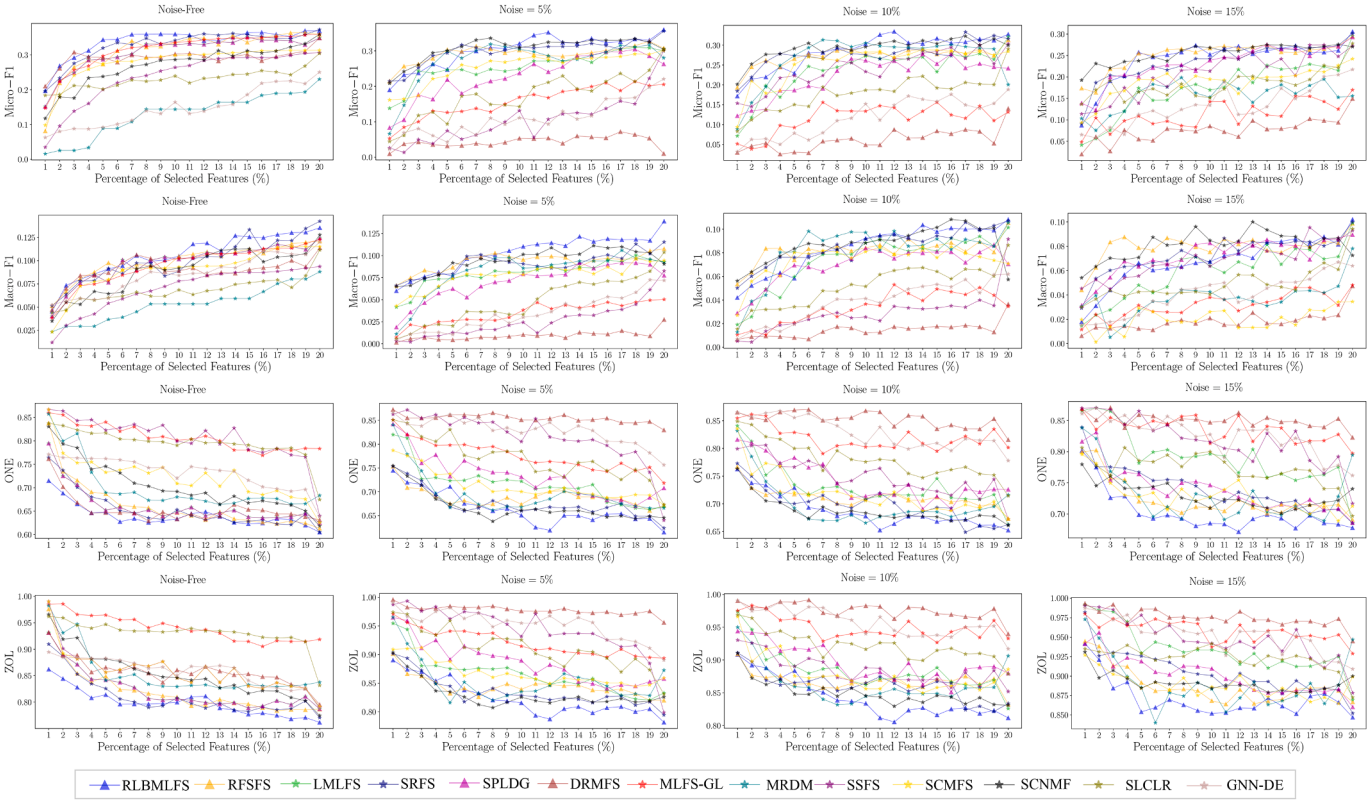


Fig. 7. Robustness analysis on Education dataset in terms of Micro-F1, Macro-F1, One-Error, and Zero-One Loss.

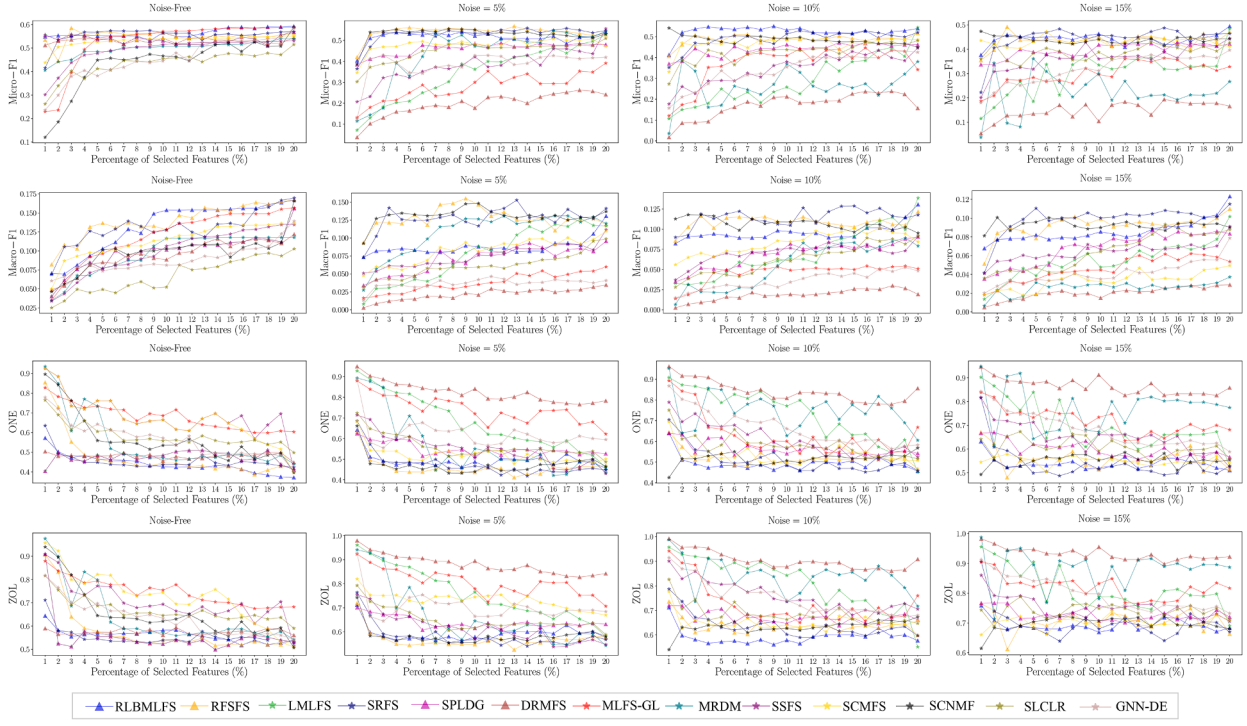


Fig. 8. Robustness analysis on Social dataset in terms of Micro-F1, Macro-F1, One-Error, and Zero-One Loss.

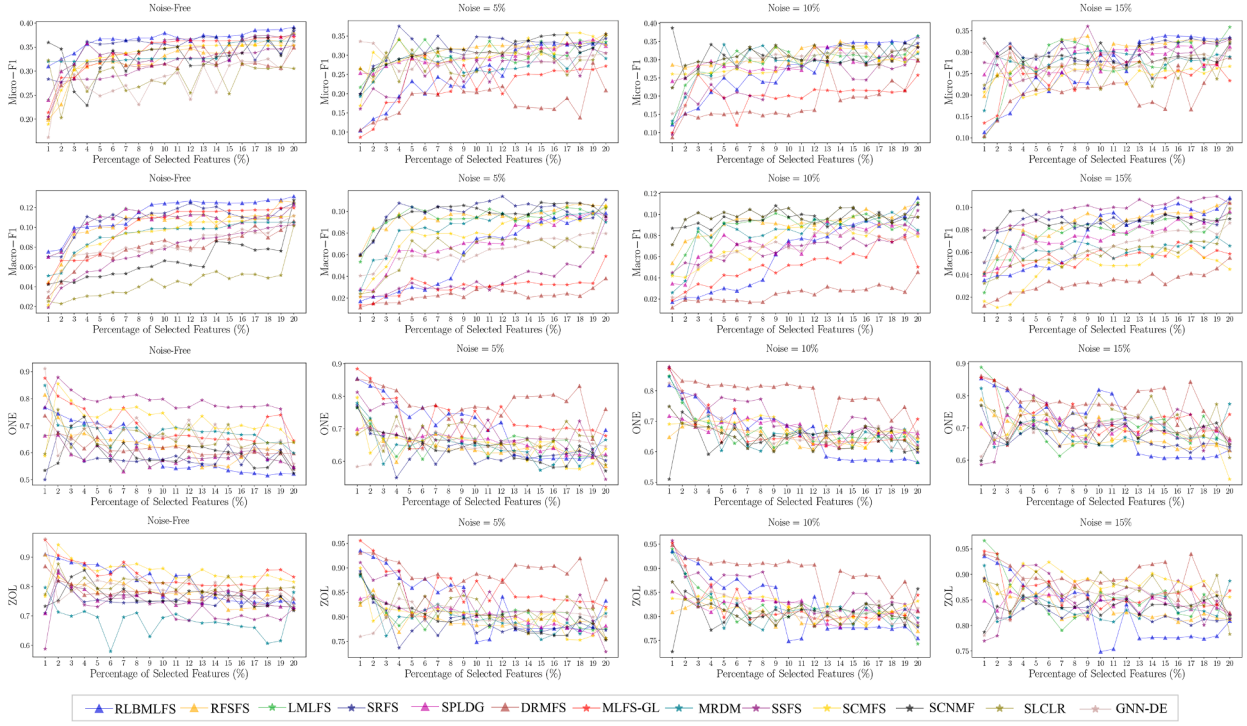


Fig. 9. Robustness analysis on Society dataset in terms of Micro-F1, Macro-F1, One-Error, and Zero-One Loss.

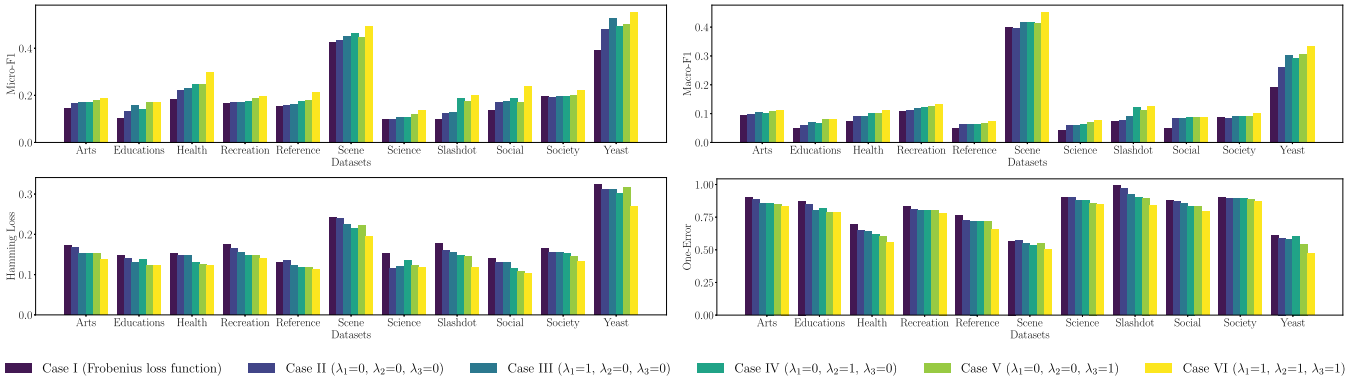


Fig. 10. Ablation study on the RLBMLFS model under 30% noise in terms of Micro-F1, Macro-F1, Hamming Loss, and One-Error.

tion of the top-k features. The detailed solution for the RLBMLFS model is presented in Algorithm 1.

3.6. Convergence analysis

In this subsection, we prove that the objective function is non-increasing under the update rule of $\boldsymbol{W}$ in Eq. (35). The proof follows the auxiliary-function (majorization-minimization) technique.

Definition 1 (Auxiliary function). A function $G(w, w')$ is an auxiliary function of $F(w)$ if it satisfies

$$G(w, w') \geq F(w) \quad \text{and} \quad G(w, w) = F(w).$$

Lemma 1. *If $G(w, w')$ is an auxiliary function of $F(w)$ and*

$$w^{t+1} = \arg\min_{w} G(w, w^t),$$

then $F(w^{t+1}) \leq F(w^t)$.

Proof. *By Definition 1,*

$$F(w^{t+1}) \leq G(w^{t+1}, w^t) \leq G(w^t, w^t) = F(w^t).$$

□

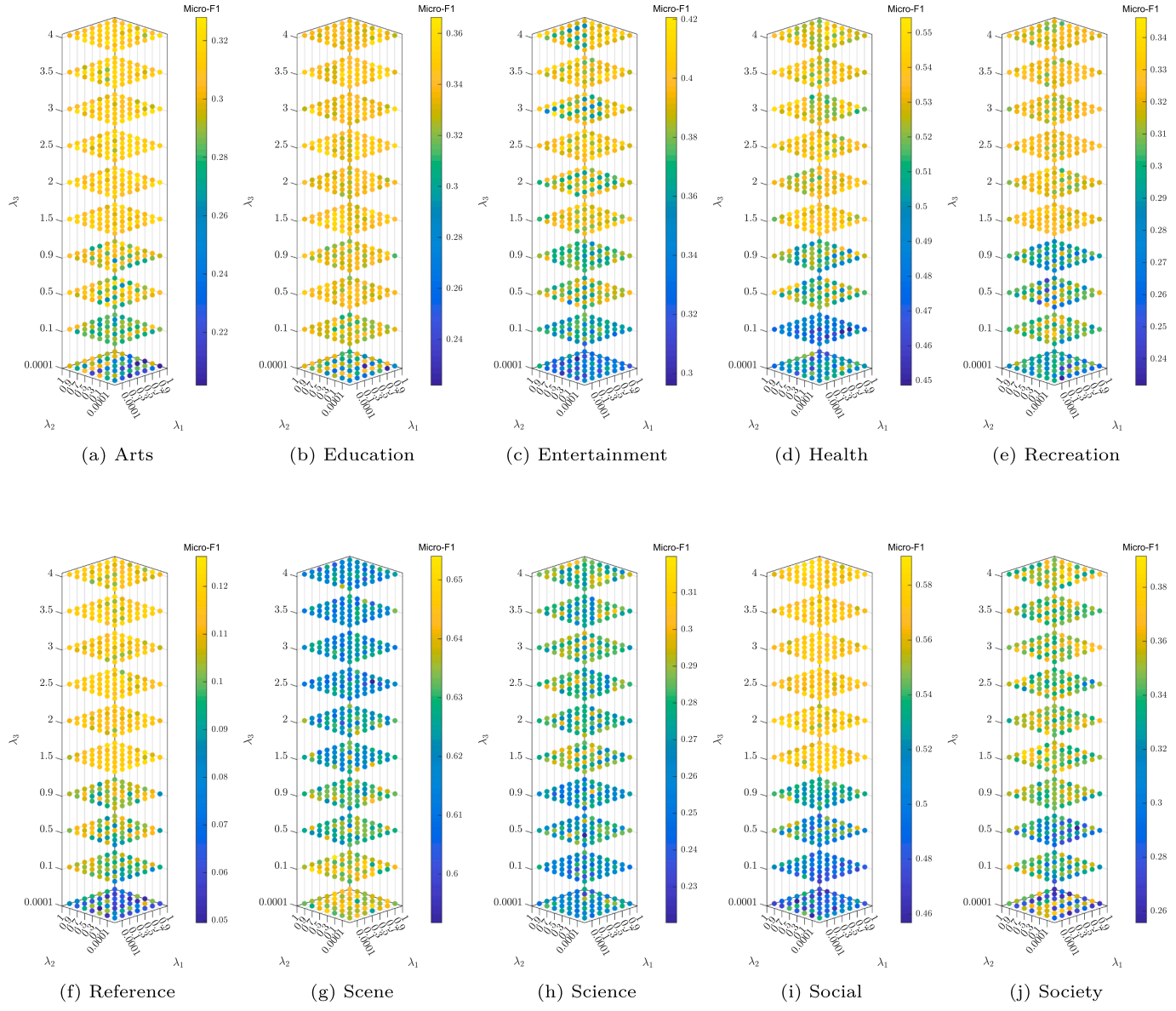
We now consider the subproblem of updating $\boldsymbol{W}$ with $\boldsymbol{C}$ (thus $\boldsymbol{L} = \boldsymbol{A} - \boldsymbol{C}$), $\boldsymbol{Z}$, and $\boldsymbol{D}$ fixed. Define

$$\begin{aligned} F(\boldsymbol{W}) = &\|\boldsymbol{Z} \circ (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})\|_F^2 + \lambda_1 \mathrm{Tr}(\boldsymbol{W}\boldsymbol{L}\boldsymbol{W}^\top) + \lambda_3 \mathrm{Tr}(\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{W}) \\ &+ \lambda_2 \sum_{i,j} \log\left(1 + (\boldsymbol{W}\boldsymbol{W}^\top)_{ij}\right) - \lambda_2 \|\boldsymbol{W}\|_F^2, \end{aligned} \tag{39}$$

with the constraint $\boldsymbol{W} \geq 0$.

Lemma 2 (Majorization of the log-term). *Let*

$$\boldsymbol{S}^t = \frac{1}{1 + \boldsymbol{W}^t (\boldsymbol{W}^t)^\top},$$



(a) Arts (b) Education (c) Entertainment (d) Health (e) Recreation

(f) Reference (g) Scene (h) Science (i) Social (j) Society

Fig. 11. Parameter analysis in term of Micro-F1 measure.

Algorithm 1 Robust log-based multi-label feature selection with dynamic label correlation and relevance-redundancy optimization (RLBMLFS).

Input: Feature matrix $\boldsymbol{X}\in\mathbb{R}^{n\times m}$ and Label matrix $\boldsymbol{Y}\in\mathbb{R}^{n\times l}$, Regularization parameters $\lambda_1, \lambda_2, \lambda_3$;
Output: Feature score $s_h = \|\boldsymbol{w}^{(h)}\|, \forall h \in \{1, 2, \ldots, m\}$.

Initialize label correlation $\boldsymbol{C}$ by (22) and $\boldsymbol{W}$ randomly;
$t = 0$;
while $t <$ MaxIteration **do**
  $E_{ij} = |Y_{ij} - [XW]_{ij}|$
  Update $Z_{ij} \leftarrow \frac{1}{(E_{ij}(1+E_{ij})+\epsilon)}$;
  Update $D_{hh} \leftarrow \frac{1}{(\|\boldsymbol{w}^{(h)}\|(1+\|\boldsymbol{w}^{(h)}\|)+\epsilon)}$;
  Update feature weight $\boldsymbol{W}$ by (35);
  Update label correlation $\boldsymbol{C}$ by (38);
  $t = t + 1$;
end while
Return $\boldsymbol{W}$;
Evaluate the feature score by $s_h = \|\boldsymbol{w}^{(h)}\|$.

where the fraction is element-wise. Then for any $\boldsymbol{W} \geq 0$,

$$\sum_{i,j} \log\left(1 + (\boldsymbol{W}\boldsymbol{W}^\top)_{ij}\right) \leq \sum_{i,j} \left[\log\left(1 + (\boldsymbol{W}^t(\boldsymbol{W}^t)^\top)_{ij}\right) + S^t_{ij}\left((\boldsymbol{W}\boldsymbol{W}^\top)_{ij} - (\boldsymbol{W}^t(\boldsymbol{W}^t)^\top)_{ij}\right)\right]. \tag{40}$$

Proof. Since $\log(\cdot)$ is concave, for any $u, v > 0$,

$$\log u \leq \log v + \frac{1}{v}(u - v).$$

Let $u = 1 + (\boldsymbol{W}\boldsymbol{W}^\top)_{ij}$ and $v = 1 + (\boldsymbol{W}^t(\boldsymbol{W}^t)^\top)_{ij}$ and sum over (i, j). □ □

Using Lemma 2, we upper-bound $F(\boldsymbol{W})$ by a surrogate function that is separable in the elements of $\boldsymbol{W}$. Specifically, define the following auxiliary function:

$$G(\boldsymbol{W}, \boldsymbol{W}^t) = \|\boldsymbol{Z} \circ (\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{W})\|_F^2 + \lambda_1 \mathrm{Tr}(\boldsymbol{W}\boldsymbol{L}\boldsymbol{W}^\top) + \lambda_3 \mathrm{Tr}(\boldsymbol{W}^\top \boldsymbol{D}\boldsymbol{W}) + \lambda_2 \mathrm{Tr}\left(\boldsymbol{W}^\top \boldsymbol{S}^t \boldsymbol{W}\right) - \lambda_2 \|\boldsymbol{W}\|_F^2 + \text{const}, \tag{41}$$

where $\boldsymbol{S}^t = 1/(1 + \boldsymbol{W}^t(\boldsymbol{W}^t)^\top)$ and "const" collects all terms independent of $\boldsymbol{W}$.

By Lemma 2, $G(\boldsymbol{W}, \boldsymbol{W}^t) \geq F(\boldsymbol{W})$ holds for all $\boldsymbol{W} \geq 0$, and clearly $G(\boldsymbol{W}^t, \boldsymbol{W}^t) = F(\boldsymbol{W}^t)$. Hence, $G(\boldsymbol{W}, \boldsymbol{W}^t)$ is an auxiliary function of $F(\boldsymbol{W})$.

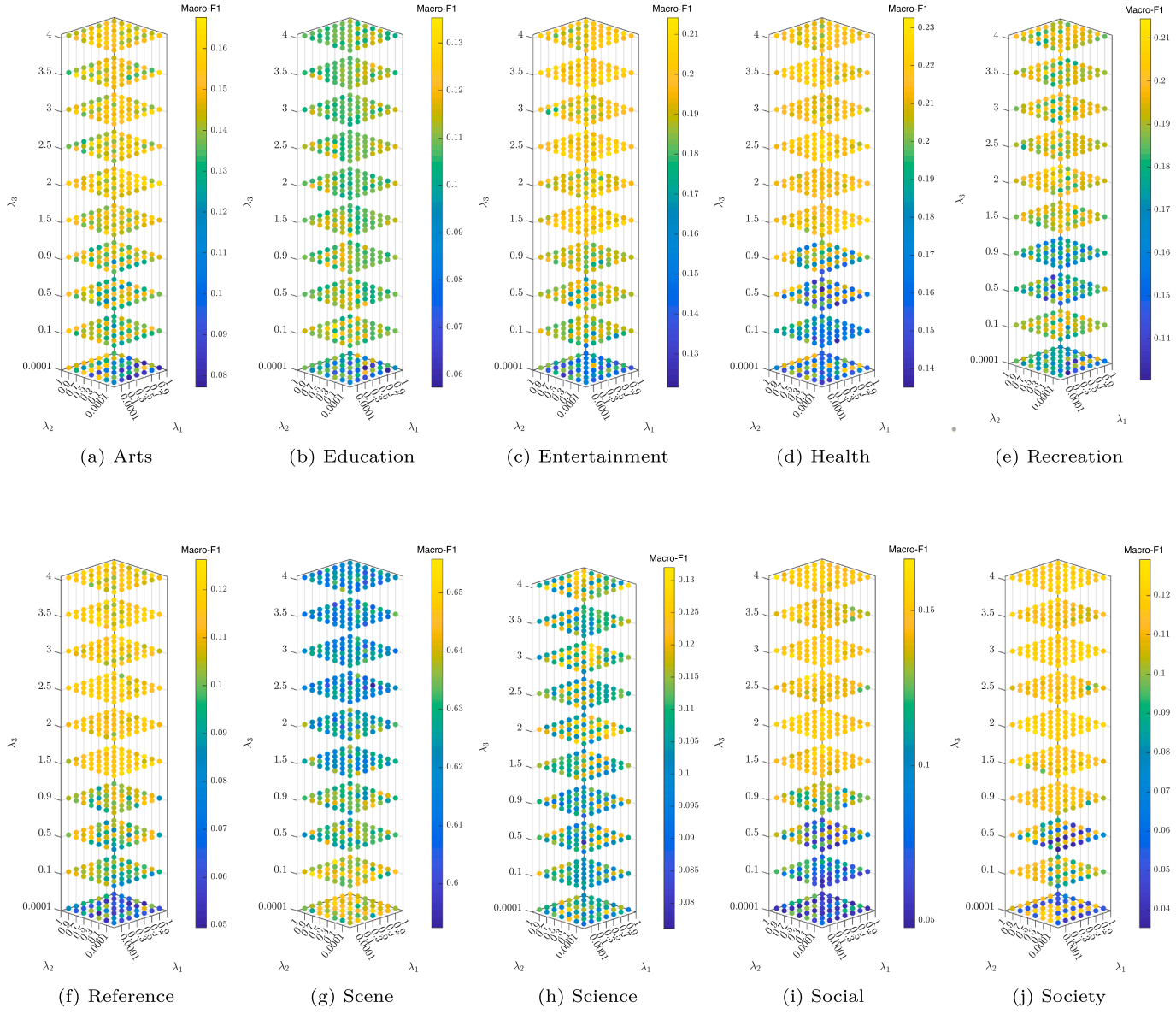


Fig. 12. Parameter analysis in term of Macro-F1 measure.

Next, we minimize $G(\boldsymbol{W}, \boldsymbol{W}^t)$ w.r.t. $\boldsymbol{W}$ under the nonnegativity constraint using the KKT conditions. Taking the derivative of G gives

$$\nabla_{\boldsymbol{W}} = -2\boldsymbol{X}^\top(\boldsymbol{Z}\circ\boldsymbol{Y}) + 2\boldsymbol{X}^\top(\boldsymbol{Z}\circ(\boldsymbol{X}\boldsymbol{W})) + 2\lambda_1\boldsymbol{W}\boldsymbol{A} - 2\lambda_1\boldsymbol{W}\boldsymbol{C} + 2\lambda_2\boldsymbol{S}^t\boldsymbol{W} - 2\lambda_2\boldsymbol{W} + 2\lambda_3\boldsymbol{D}\boldsymbol{W}. \tag{42}$$

Let $\boldsymbol{\Omega} \geq 0$ be the Lagrange multiplier for $\boldsymbol{W} \geq 0$. The KKT condition yields $\boldsymbol{\Omega}\circ\boldsymbol{W} = 0$ and $\nabla_{\boldsymbol{W}} - \boldsymbol{\Omega} = \boldsymbol{0}$. Using element-wise complementary slackness leads to the multiplicative update

$$\boldsymbol{W}^{t+1} = \boldsymbol{W}^t \circ \frac{\boldsymbol{X}^\top(\boldsymbol{Z}\circ\boldsymbol{Y}) + \lambda_1\boldsymbol{W}^t\boldsymbol{C} + \lambda_2\boldsymbol{W}^t}{\boldsymbol{X}^\top(\boldsymbol{Z}\circ(\boldsymbol{X}\boldsymbol{W}^t)) + \lambda_1\boldsymbol{W}^t\boldsymbol{A} + \lambda_2\boldsymbol{S}^t\boldsymbol{W}^t + \lambda_3\boldsymbol{D}\boldsymbol{W}^t}, \tag{43}$$

which is identical to Eq. (35) since $\boldsymbol{S}^t = 1/(1 + \boldsymbol{W}^t(\boldsymbol{W}^t)^\top)$.

Theorem 1. *Under the update rule in Eq. (43), the objective in Eq. (39) is non-increasing, i.e.,*

$F(\boldsymbol{W}^{t+1}) \leq F(\boldsymbol{W}^t)$.

Proof. We have shown that $G(\boldsymbol{W}, \boldsymbol{W}^t)$ is an auxiliary function of $F(\boldsymbol{W})$ and that $\boldsymbol{W}^{t+1}$ minimizes $G(\boldsymbol{W}, \boldsymbol{W}^t)$ under $\boldsymbol{W} \geq 0$ via Eq. (43). Therefore, by Lemma 1,

$F(\boldsymbol{W}^{t+1}) \leq F(\boldsymbol{W}^t)$.

Since the objective function in (36) is convex with respect to $\boldsymbol{C}$ when $\boldsymbol{W}$ is fixed, the multiplicative update in Eq. (38) satisfies the KKT conditions and guarantees a non-increasing objective value. Therefore, the optimization with respect to $\boldsymbol{C}$ converges. □

3.7. Complexity analysis

We analyze the computational cost of the proposed RLBMLFS optimization by deriving the per-iteration time and memory complexities of the alternating updates for the feature–label relevancy matrix $\boldsymbol{W}$ and the label correlation matrix $\boldsymbol{C}$. Let n denote the number of instances, d the number of features, l the number of labels, with $\boldsymbol{X} \in \mathbb{R}^{n\times d}$, $\boldsymbol{Y}, \boldsymbol{Z} \in \mathbb{R}^{n\times l}$, $\boldsymbol{W} \in \mathbb{R}^{d\times l}$, and $\boldsymbol{C} \in \mathbb{R}^{l\times l}$. In each outer iteration, computing the reconstruction $\boldsymbol{X}\boldsymbol{W}$ to form the error and weights $\boldsymbol{Z}$ costs $\mathcal{O}(ndl)$ (plus $\mathcal{O}(nl)$ elementwise operations). The $\boldsymbol{W}$-update in Eq. (35) is dominated by the two data-fitting products $\boldsymbol{X}^\top(\boldsymbol{Z}\circ\boldsymbol{Y})$ and $\boldsymbol{X}^\top(\boldsymbol{Z}\circ(\boldsymbol{X}\boldsymbol{W}))$, each requiring $\mathcal{O}(ndl)$, the correlation terms $\boldsymbol{W}\boldsymbol{C}$ and $\boldsymbol{W}\boldsymbol{A}$ with cost $\mathcal{O}(dl^2)$ (reduced to $\mathcal{O}(dl)$ if $\boldsymbol{A}$ is diagonal), and the redundancy-related computation of $\boldsymbol{W}\boldsymbol{W}^\top$ induced by the log penalty, which costs $\mathcal{O}(d^2l)$ (followed by only lower-order $\mathcal{O}(d^2)$ elementwise operations). The sparsity term $\boldsymbol{D}\boldsymbol{W}$ and

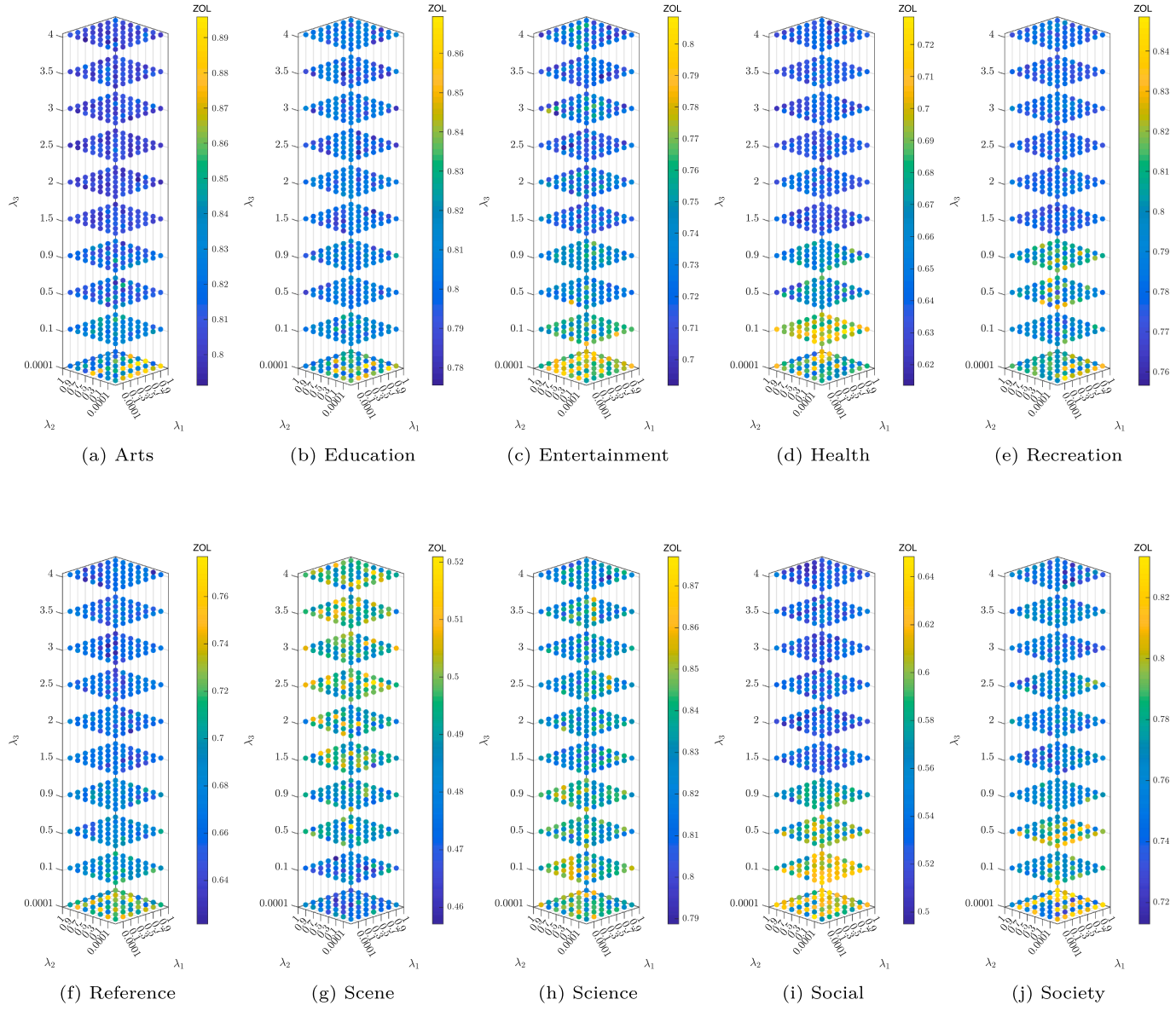


Fig. 13. Parameter analysis in term of Zero-One Loss measure.

updating diagonal $\boldsymbol{D}$ add $\mathcal{O}(dl)$. The $\boldsymbol{C}$-update in Eq. (38) mainly requires forming $\boldsymbol{W}^\top\boldsymbol{W} \in \mathbb{R}^{l\times l}$, which costs $\mathcal{O}(dl^2)$, followed by $\mathcal{O}(l^2)$ normalization and elementwise updates. Therefore, the per-iteration time complexity is $\mathcal{O}(ndl + dl^2 + d^2l)$ and the total time over T iterations is $\mathcal{O}\big(T(ndl + dl^2 + d^2l)\big)$ (or $\mathcal{O}\big(T(sl + dl^2 + d^2l)\big)$ when $\boldsymbol{X}$ is sparse with s non-zeros). In terms of memory, storing $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ requires $\mathcal{O}(nd + nl)$, storing $\boldsymbol{W}$ requires $\mathcal{O}(dl)$, storing $\boldsymbol{C}$ (and optionally $\boldsymbol{W}^\top\boldsymbol{W}$) requires $\mathcal{O}(l^2)$, storing diagonal $\boldsymbol{D}$ requires $\mathcal{O}(d)$, and explicitly materializing $\boldsymbol{W}\boldsymbol{W}^\top$ incurs an additional $\mathcal{O}(d^2)$, yielding an overall memory complexity of $\mathcal{O}(nd + nl + dl + l^2 + d^2)$ (or $\mathcal{O}(nd + nl + dl + l^2)$ if $\boldsymbol{W}\boldsymbol{W}^\top$ is computed blockwise and not stored). In summary, the dominant cost depends on whether the dataset is sample-heavy (typically ndl) or high-dimensional (typically d^2l), and exploiting sparsity in $\boldsymbol{X}$ and avoiding explicit storage of $\boldsymbol{W}\boldsymbol{W}^\top$ can substantially reduce the practical runtime and memory footprint.

4. Experimental study

In this section, we present a comprehensive evaluation of the RLBMLFS method. To assess its effectiveness, we conduct a series of experiments comparing RLBMLFS with 13 state-of-the-art techniques: SCLS [27], SCMFS [3], SSFS [21], MRDM [22], MLFS-GL [13], LMLFS [33], DRMFS [9], RFSFS [14], SCNMF [30], SLCLR [31], SPLDG [46], SRFS [23], and GNN-DE [34]. The comparison is carried out using multiple quantitative performance metrics to provide a clear and comprehensive assessment of RLBMLFS. Furthermore, we conduct an in-depth analysis of RLBMLFS from several critical perspectives. Experiments present the tabulated results along with radar-chart visualizations. We also perform ablation studies to isolate the contributions of key components, parameter sensitivity experiments to evaluate the effect of parameter variations, an experimental comparison between static and learned label correlations, and convergence tests to assess stability. Finally, we analyze the running time of RLBMLFS to assess its computational efficiency across different datasets.

4.1. Datasets

During our experiments, we evaluated our model on 17 datasets from the Mulan Library's multi-label text and image classification collection. The Yahoo multi-label datasets, introduced by Yahoo Labs, are

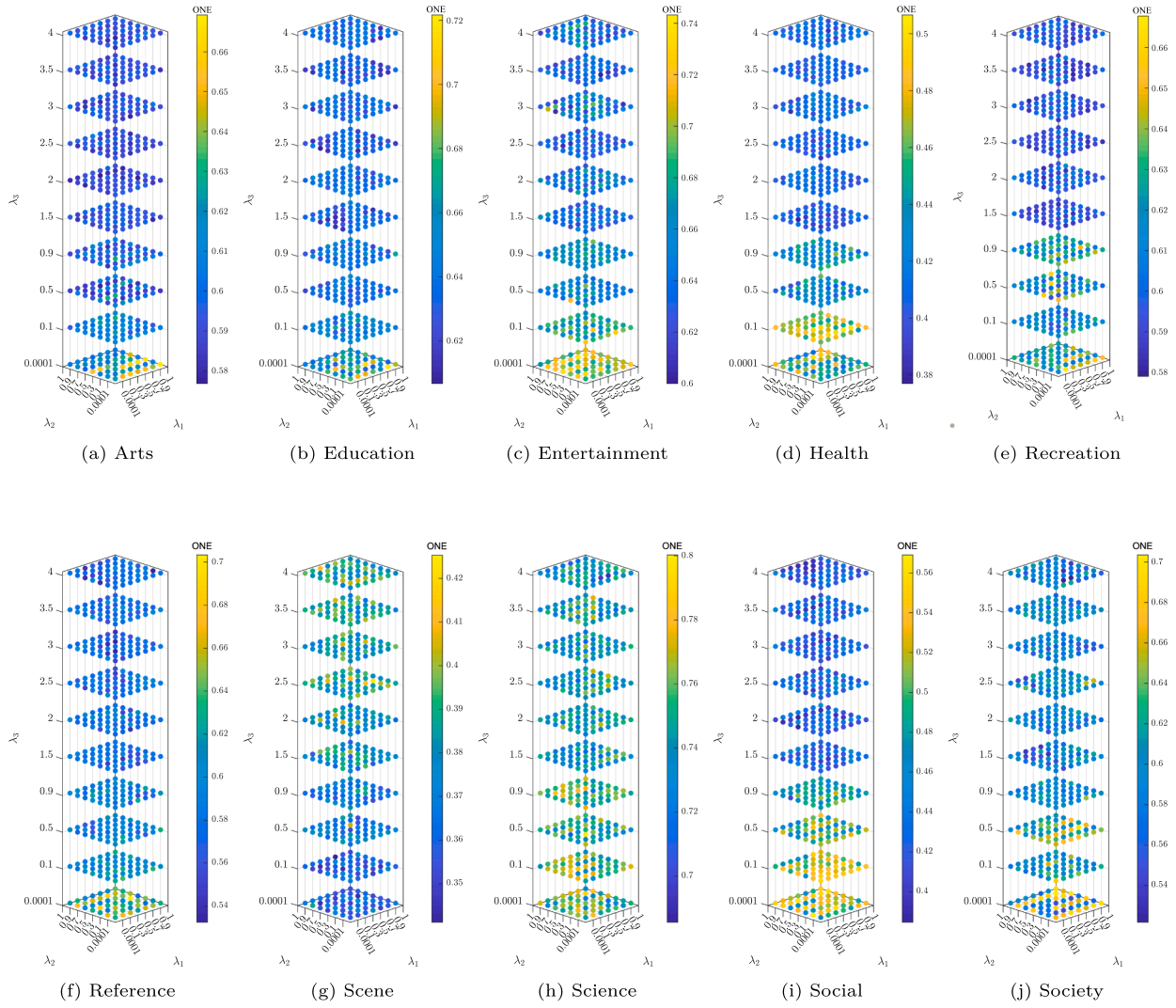


Fig. 14. Parameter analysis in term of One-Error measure.

widely used for multi-label classification tasks. These datasets contain a vast collection of text documents, images, and music annotated with multiple labels, with each dataset comprising a 40% training set and a 60% test set of instances [47]. Additionally, we included the multi-label multi-view NUS-WIDE dataset [48], which contains naturally noisy, real-world multi-label annotations derived from crowd-sourced tags and user-provided metadata. These annotations inherently include label noise due to subjectivity and inconsistencies in human labeling. These benchmark datasets are commonly employed in research to assess the performance of multi-label classification algorithms. Table 2 presents a detailed overview of each dataset.

Table 2
Detailed information of the real-world datasets.

Dataset	Training	Test	#Instances	#Features	#Labels	Domain
Arts	2000	3000	5000	462	26	Text
Educations	2000	3000	5000	550	33	Text
Emotions	237	356	593	72	6	Music
Entertainment	2000	3000	5000	640	21	Text
Health	2000	3000	5000	612	32	Text
Image	800	1200	2000	294	5	Image
Languagelog	584	875	1459	1004	75	Text
Medical	391	587	978	1449	45	Text
Recreation	2000	3000	5000	606	22	Text
Reference	2000	3000	5000	793	33	Text
Scene	963	1444	2407	294	6	Image
Science	2000	3000	5000	743	40	Text
Slashdot	1513	2269	3782	1079	22	Text
Social	2000	3000	5000	1047	39	Text
Society	2000	3000	5000	636	27	Text
Yeast	967	1450	2417	103	14	Biology
NUS-WIDE	7211	7275	14486	634	63	Image

4.2. Evaluation metrics

To evaluate the performance of all competing methods, we establish the Multi-Label k-Nearest Neighbors (ML-kNN) algorithm [49] as the benchmark classifier. ML-kNN is widely utilized in multi-label feature selection approaches for classification tasks [3,25,50,51]. We set k = 10 for the number of nearest neighbors. Additionally, we employ five commonly used evaluation metrics: Micro-F1 (MIC), Macro-F1 (MAC), Hamming Loss (HML), Zero-One Loss (ZOL), and One-Error (ONE) [52, 53]. Lower values for Hamming Loss, Zero-One Loss, One-Error indicate better classification performance, while higher values for Micro-F1, and Macro-F1 reflect improved results.

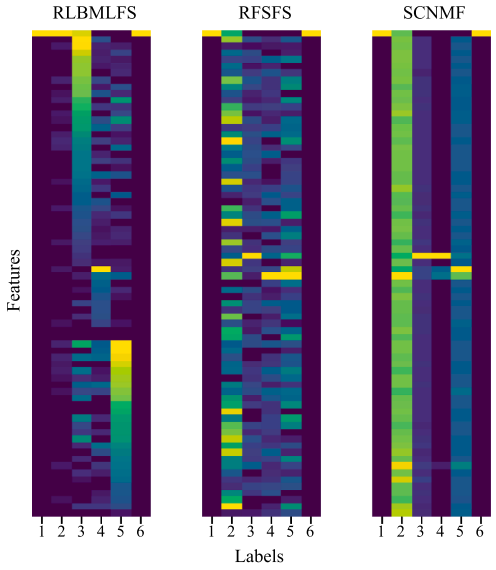


Fig. 15. Feature-label importance matrices on Emotions (72 features, 6 labels) for RLBMLFS, RFSFS, and SCNMF (brighter = more important). Features are grouped as MeanAcc-MeanMem (1–16), MeanAcc-StdMem (17–32), StdAcc-MeanMem (33–48), StdAcc-StdMem (49–64), and BeatHistogram (65–72).

4.3. Compared methods

We evaluate our method by comparing it with 13 well-known state-of-the-art multi-label feature selection techniques. A brief description of each method is provided below.

- **SCLS** [27] is an MLFS algorithm that leverages information theory to assess conditional relevance through a scalable relevance evaluation criterion.
- **SCMFS** [3] employs coupled nonnegative matrix factorization to create the shared common model between features and labels.
- **SSFS** [21] utilizes Latent Structure Shared (LSS) and Learning Label Specific (LLS) to capture the latent feature structure in the original feature space and ensure similar features share similar labels.
- **MRDM** [22] employs the Hilbert-Schmidt Independence Criterion (HSIC) to enhance the dependence between the manifold embedding and the class labels.
- **MLFS-GL** [13] leverages global-local correlations to capture explicit label correlation, and uses graph regularization to maintain consistency between the original and latent spaces.
- **LMLFS** [33] utilizes the $L_{2,1}$-norm on the self-representation matrix to select informative features and reduce the effects of noise.
- **DRMFS** [9] uses $L_{2,1}$-norm loss function to increase robustness against noisy labels and incorporated the graph regularizations of labels and features to enhance the quality of predicted labels.
- **RFSFS** [14] uses global label correlation with L_1-norm and L_2-norm to capture the relationships between the labels and reduce redundancy among the features respectively.
- **SCNMF** [30] enhances similarity preservation by constraining the inner products of the feature weight matrix and exploits label similarity to improve label weight learning.
- **SLCLR** [31] applies the L_1-norm to eliminate irrelevant features and incorporates the L_1-norm and $L_{2,1}$-norm to remove redundant features. In addition, it introduces a Laplacian rank constraint to enhance sparsity.
- **SPLDG** [46] imposes detailed constraints on pseudo-labels, pays closer attention to specific labels, and leverages the feature and dynamic manifolds to constrain feature weights.
- **SRFS** [23] employs a sparse supplementation term with the $L_{2,1}$-norm as a fusion component to enhance the sparsity of feature selection.
- GNN-DE [34] combines GNN to model feature-label dependencies with Differential Evolution to efficiently optimize the feature subset, enabling effective multi-label feature selection.

4.4. Experimental results

In this experiment, we evaluated the average performance of each method by selecting the top 20% of features. The results, measured across five evaluation metrics, are presented in Tables 3–7, encompassing noise levels of 15%, 30%, 40%, and 50%. For each dataset, the best results are highlighted in bold, with higher values representing better classification performance. To ensure reliable evaluation and minimize the impact of random initialization, each method was run 5 times, and the reported results are averages across all datasets [54]. Moreover, to substantiate the robustness of our approach in real-world noisy environments, we extended the evaluation to include the NUS-WIDE dataset, which inherently contains realistic noisy labels. The results of this additional analysis are presented in Table 8. The results show that the proposed model consistently outperforms the others on most datasets. Furthermore, its performance improves as noise levels increase, demonstrating its robustness under challenging noisy conditions.

Additionally, to better compare our method with other approaches under 30% noise and to highlight differences in performance, we use the radar diagrams in Fig. 4 to evaluate its effectiveness across five metrics on six multi-label datasets: Arts, Recreation, Entertainment, Scene, Science, and Slashdot. These metrics assess various aspects of the quality and effectiveness of the methods [55]. It is important to highlight that the Hamming Loss, Zero-One Loss, and One-Error metrics are interpreted inversely to align with the other evaluation measures, ensuring that higher values consistently represent better performance. For fair comparisons, the data in Fig. 4 are normalized to values between 0.5 and 1. A method that covers more area in the radar charts performs better across all evaluation criteria. As illustrated, our method achieves broader coverage than competing approaches, indicating superior adaptability and performance. This visualization effectively highlights not only the strength of our approach but also the relative distance from other methods, showcasing its ability to handle diverse challenges and datasets with greater accuracy and efficiency.

To further investigate the relative performance of the proposed RLBMLFS method in comparison with other competing approaches, the Friedman test is employed on Yahoo Labs datasets. The Friedman statistic F_F and the corresponding critical values for each evaluation metric, derived according to the Iman-Davenport theory [56], are reported in Table 9. As shown in Table 9, at a significance level of 0.05, the null hypothesis stating that all compared methods exhibit identical performance is rejected. Consequently, the Bonferroni–Dunn test is applied as a post hoc analysis, in which RLBMLFS is treated as the control method. The statistical significance between the control method and the other approaches is determined by examining whether the difference in their average ranks falls within the critical distance (CD), which is computed as $\mathrm{CD} = q_\alpha \sqrt{\frac{k(k+1)}{6N}}$, where $q_\alpha = 3.01$, $\alpha = 0.05$, $k = 14$, and $N = 16$, resulting in $\mathrm{CD} = 4.90$. Fig. 5 presents the CD diagrams for all evaluation metrics. In each diagram, the red line represents the critical distance from RLBMLFS, indicating the methods whose performance differences are not statistically significant. Although no significant differences are observed between RLBMLFS and several competing methods, the proposed approach demonstrates a clear competitive advantage over most alternatives and consistently achieves the highest rank across all evaluation metrics.

To comprehensively assess the effect of varying feature selection rates on classification performance and their robustness across different noise levels, we use four datasets: Arts, Education, Social, and Society.

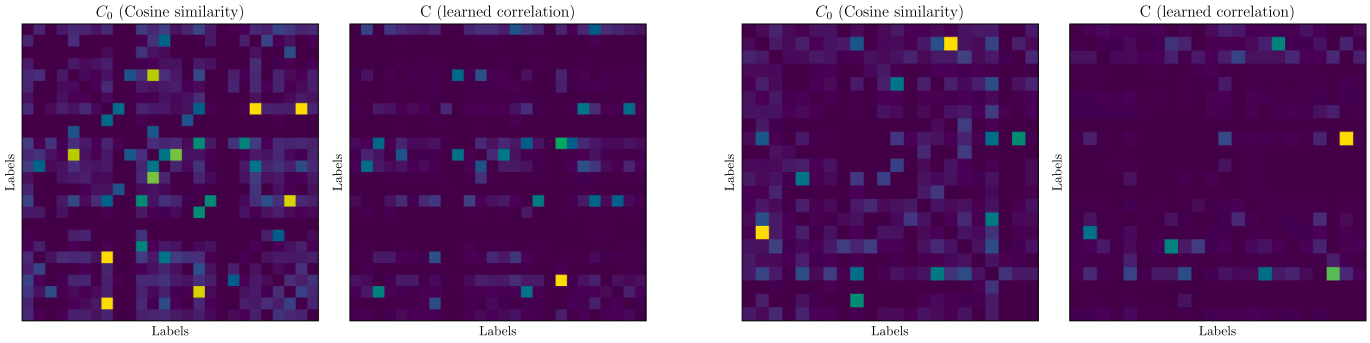


Fig. 16. Heatmaps of the cosine-similarity label correlation prior $\boldsymbol{C}_0$ and the correlation matrix $\boldsymbol{C}$ learned by our model for Arts and Recreation, showing that learning yields a sparser, higher-contrast, and more selective set of label dependencies.

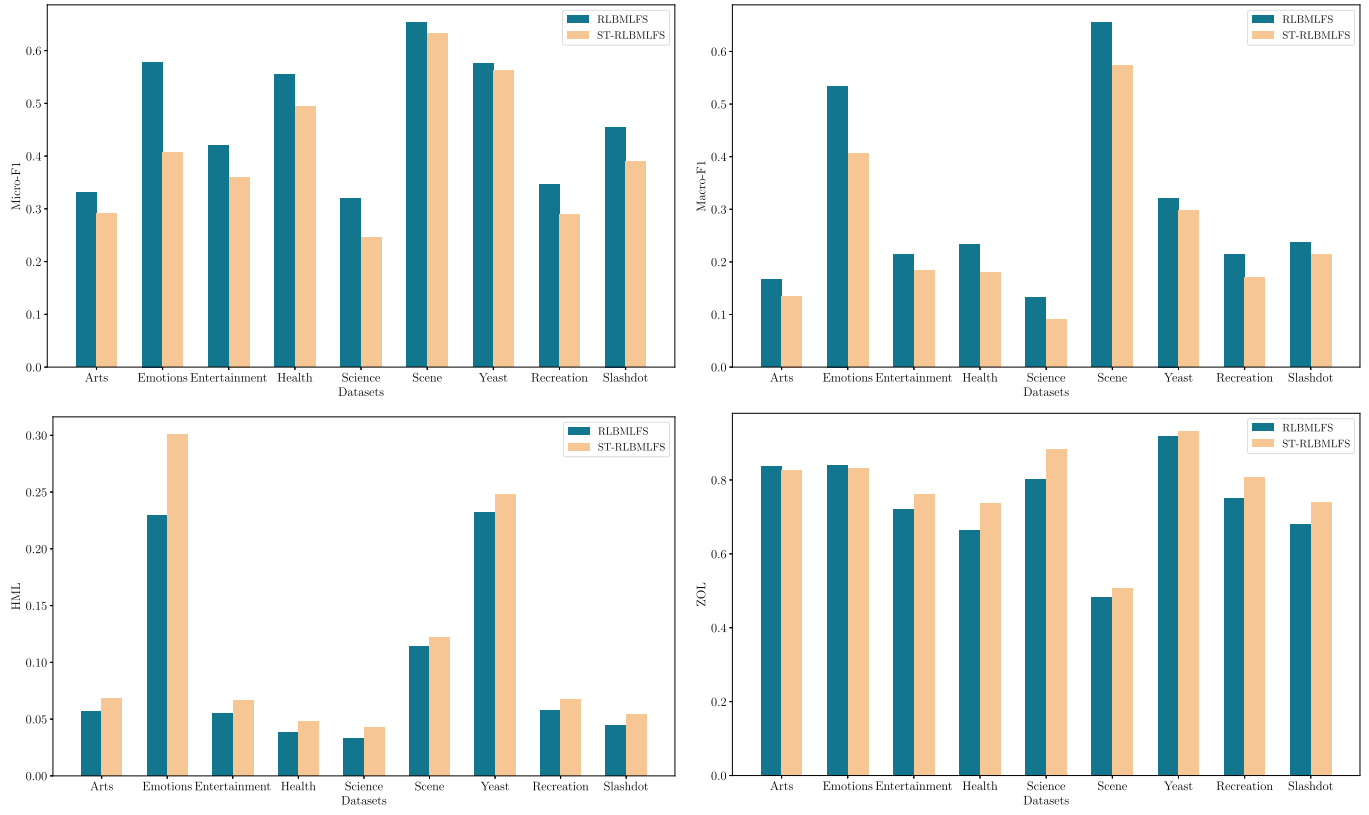


Fig. 17. Comparison of learned (RLBMLFS) and static (ST-RLBMLFS) label correlations in terms of Micro-F1, Macro-F1, HML, and ZOL measures.

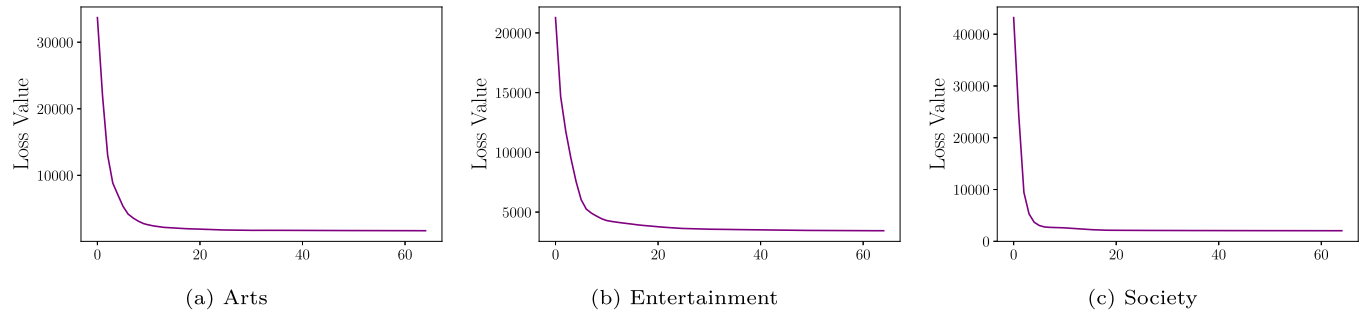


Fig. 18. Convergence analysis of RLBMLFS model on the Arts, Entertainment, and Society datasets.

We evaluate the efficacy of these selection rates in identifying the most relevant features under various noise conditions: 0% (noise-free), 5%, 10%, and 15% noise levels. In the noisy scenarios, a portion of the labels in the training set is randomly flipped, meaning that 0 and 1 are interchanged. Figs. 6–9 visually depict this analysis, illustrating how classification performance improves as the feature selection rate increases from 1% to 20%. On the y-axis, classification metrics are Micro-F1, Macro-F1, One-Error, and Zero-One Loss, while the x-axis represents the number of features selected. The observed trend shows a steady improvement in classification performance as the number of selected features increases.

4.5. Ablation study

To analyze the influence of each term, we conducted extensive ablation experiments on nine datasets. The evaluation metrics used were Micro-F1, Macro-F1, Zero-One Loss, and One-Error (Tables 10 and 11), providing a comprehensive assessment of the model's performance across various aspects of multi-label classification. The results, illustrated in Fig. 10 under 30% label noise, demonstrate performance variations across different cases. We defined six evaluation cases to isolate and highlight the effectiveness of each component of our model.

In Case I, we replaced the proposed Hx loss with a conventional Frobenius-norm loss functions and disabled all regularization terms. This configuration exhibits substantial performance drops across all evaluation metrics, highlighting its vulnerability to severe label noise.

In Case II ($\lambda_1 = 0, \lambda_2 = 0, \lambda_3 = 0$), RLBMLFS operates without any regularization terms and relies solely on the proposed robust Hx loss function. Compared to Case I, this configuration demonstrates consistently superior performance and noticeably greater stability under increasing noise levels. Most importantly, under 30% label noise, Case II significantly outperforms the Frobenius-based baseline, clearly validating the intrinsic robustness of the Hx loss. Despite this improvement, the absence of regularization still makes the model susceptible to overfitting, leaving room for further performance gains.

In Case III ($\lambda_1 = 1, \lambda_2 = 0, \lambda_3 = 0$), we integrated the dynamic label correlation regularization term into RLBMLFS. Compared with Case II, this configuration improves predictive performance by effectively capturing inter-label dependencies, which is particularly beneficial for multi-label learning. Moreover, leveraging learned label correlations in conjunction with the robust Hx loss further alleviates the adverse impact of noisy labels.

In Case IV ($\lambda_1 = 0, \lambda_2 = 1, \lambda_3 = 0$), only the logarithmic penalty term is activated. As illustrated in Fig. 10, this component effectively reduces feature redundancy and enhances the quality of the selected feature subset, leading to consistent improvements in Micro-F1 and Macro-F1 scores.

In Case V ($\lambda_1 = 0, \lambda_2 = 0, \lambda_3 = 1$), we incorporated the sparsity-inducing regularization term into the model. This setting yields clear performance gains across all evaluation measures. The results indicate that enforcing sparsity not only mitigates overfitting but also improves generalization by forcing the model to focus on the most informative features.

Finally, in Case VI ($\lambda_1 = 1, \lambda_2 = 1, \lambda_3 = 1$), we implemented the complete RLBMLFS model by combining dynamic label correlation, logarithmic penalty, and sparsity regularization. As expected, this comprehensive configuration achieves the best overall performance across all metrics. The results demonstrate the strong complementary and synergistic effects of the proposed components.

Based on the above analysis, we conclude that the proposed robust Hx loss provides a substantial robustness advantage over the conventional Frobenius-based formulation, particularly under high-noise conditions such as the 30% label noise setting, while the regularization components further enhance both robustness and generalization performance.

4.6. Analyzing the sensitivity of hyperparameters

In this section, we analyze the influence of three parameters: the dynamic label correlation (λ_1), the logarithmic penalty (λ_2), and the sparsity regularization term (λ_3). Our experiments explored various combinations of these parameters within the ranges $\lambda_1, \lambda_2 \in \{0.0001, 0.1, 0.3, 0.5, 0.7, 0.9, 1\}$ and $\lambda_3 \in \{0.0001, 0.1, 0.5, 0.9, 1.5, 2, 2.5, 3, 3.5, 4\}$ to assess their impact on model performance. These evaluations were conducted across 10 datasets, using four evaluation metrics: Micro-F1, Macro-F1, Zero-One Loss, and One-Error.

The results are visualized through three-dimensional plots in Figs. 11–14, where the x-axis represents λ_1, the y-axis represents λ_2, and the z-axis represents λ_3. In these Figures, pale yellow colors indicate improved performance for Micro-F1 and Macro-F1, whereas blue colors correspond to better results for Zero-One Loss and One-Error. Overall, the visual trends demonstrate that appropriate increases in the parameter values generally lead to improved classification performance, particularly within specific stable ranges. Specifically, higher λ_1 enhances dynamic label correlation, enabling better exploitation of inter-label dependencies; λ_2 effectively suppresses irrelevant information, resulting in more reliable predictions; and λ_3 promotes sparsity, reducing redundancy and improving generalization. The parameter sensitivity analysis indicates that λ_3 is the most influential hyperparameter, showing a strong positive correlation with measures. Performance improves rapidly as λ_3 increases from 10^{-4} and reaches a stable optimum in the range $\lambda_3 \in [1.0, 3.5]$, while very small values ($\lambda_3 \leq 10^{-2}$) lead to substantial performance degradation. In contrast, λ_1 exhibits low-to-moderate sensitivity: the performance improves when λ_1 increases from 10^{-4} and saturates for $\lambda_1 \in [10^{-3}, 10^{-2}]$, with negligible gains beyond this interval. Finally, λ_2 is the least sensitive parameter; once λ_2 exceeds 10^{-3}, the performance remains largely stable, making $\lambda_2 \in [10^{-3}, 10^{-2}]$ an efficient and robust choice. Overall, optimal performance is primarily driven by λ_3, while λ_1 and λ_2 can be fixed within their respective stable ranges without significant impact.

4.7. Qualitative analysis

We conduct a qualitative comparison on the Emotions multi-label music dataset, where each sample is represented by 72 audio descriptors and annotated with 6 emotion labels, (1) amazed-surprised, (2) happy-pleased, (3) relaxing-calm, (4) quiet-still, (5) sad-lonely, and (6) angry-aggressive. The descriptors include interpretable spectral measures (spectral centroid, rolloff, and flux) computed under different mean/standard-deviation statistics, beat/tempo-related features derived from beat histograms (e.g., peak BPM/energy and summary statistics), and a set of cepstral/timbral coefficients (MFCCs) that compactly capture the spectral envelope. We compare our method, RLBMLFS, against RFSFS and SCNMF, which we select as baselines because they are more recent methods and provide a comparable multi-label feature selection setting. Fig. 15 visualizes the learned feature-label importance matrices for each method, where brighter entries indicate more influential features for a given label. Overall, RLBMLFS yields a sparse but highly structured importance pattern: strong weights concentrate in compact, label-specific subsets of features, forming clear blocks with limited cross-label leakage. This behavior suggests that RLBMLFS discovers more discriminative and interpretable feature-label associations, rather than distributing moderate weights broadly. In contrast, RFSFS exhibits a more diffuse, speckled pattern across labels, implying that many selected features contribute weakly to multiple emotions and thus provide less label-specific explanation. SCNMF shows an even stronger tendency toward feature reuse, with importance dominated by one or two label columns and fewer clearly separated feature subsets, which reduces interpretability and indicates weaker differentiation between emotion categories. As a concrete example, for the quiet-still label, RLBMLFS highlights a localized cluster of high-importance features (a distinct block in the corre-

Table 3
Micro-F1 scores on real-world datasets under different label noise levels: 0% (noise-free), 15%, 30%, 40%, and 50%. The highest score in each row is shown in **bold**, and the second-highest in underlined.

Dataset	Noise	RLBMLFS	SRFS	RFSFS	LMLFS	SPLDG	DRMFS	SSFS	SCMFS	MLFS-GL	MRDM	SCLS	SCNMF	SLCLR	GNN-DE
	0%	**0.3305**	0.3294	0.3289	0.3135	0.2383	0.2578	0.2263	0.2674	0.3230	0.2496	0.1258	0.3127	0.2123	0.2306
	15%	**0.2833**	0.2689	0.2691	0.2407	0.2369	0.2105	0.2435	0.1988	0.2104	0.1587	0.1794	0.2422	0.2109	0.1954
Arts	30%	**0.1881**	0.1694	0.1641	0.1668	0.1667	0.1110	0.1642	0.1409	0.1474	0.1143	0.1303	0.1738	0.1662	0.1802
	40%	**0.1576**	0.1390	0.1459	0.1437	0.1357	0.1166	0.1410	0.1468	0.1299	0.1391	0.1358	0.1371	0.1422	0.1327
	50%	0.1273	0.1197	0.1242	0.1237	0.1190	0.1272	0.1218	0.1194	0.1138	0.1124	0.1113	**0.1276**	0.1248	0.1145
	0%	0.3691	**0.3725**	0.3528	0.3485	0.3001	0.3330	0.3061	0.3131	0.3594	0.2306	0.1383	0.3589	0.3042	0.2504
	15%	**0.3058**	0.3010	0.2949	0.2953	0.2928	0.1494	0.2724	0.2421	0.1698	0.1557	0.1354	0.2703	0.2780	0.2171
Educat	30%	**0.1736**	0.1615	0.1592	0.1484	0.1489	0.1100	0.1579	0.1076	0.1218	0.0864	0.1111	0.1423	0.1662	0.1589
	40%	**0.1254**	0.1155	0.1139	0.1152	0.1150	0.0974	0.1143	0.1102	0.1011	0.0945	0.0890	0.1136	0.1198	0.1028
	50%	0.0948	0.0912	0.0900	0.0905	0.0902	0.0861	0.0889	0.0906	0.0882	0.0890	0.0876	0.0939	**0.0987**	0.0855
	0%	**0.5784**	0.4285	0.4242	0.3209	0.3626	0.3389	0.3209	0.3210	0.3210	0.3604	0.3604	0.3426	0.3650	0.3804
	15%	**0.5456**	0.3398	0.4664	0.3760	0.4017	0.3907	0.3673	0.3606	0.3210	0.3604	0.3604	0.3855	0.4140	0.3801
Emotio	30%	0.4731	0.4464	0.4496	0.4504	0.4314	0.3880	0.4577	0.2665	0.2740	0.3604	0.3556	0.3378	**0.4795**	0.3911
	40%	0.4347	0.3962	0.4056	0.4249	0.4542	0.3418	0.4158	0.3775	0.3590	0.2275	0.4569	0.3691	**0.4714**	0.3787
	50%	0.4734	0.4428	0.4646	0.4517	0.4597	0.4558	0.4688	0.4727	0.4223	0.3715	0.4273	0.4563	**0.4735**	0.4714
	0%	0.4208	0.4252	0.4213	0.4049	0.3271	0.1592	0.3314	0.3640	**0.4377**	0.3597	0.2665	0.4302	0.3232	0.3683
	15%	**0.3659**	0.3384	0.3504	0.3263	0.3044	0.1684	0.3170	0.2906	0.2322	0.2499	0.2324	0.3104	0.2916	0.2660
Entert	30%	**0.2487**	0.2397	0.2369	0.2353	0.2193	0.1540	0.2315	0.2001	0.1851	0.1655	0.1487	0.2313	0.2405	0.2458
	40%	**0.1737**	0.1554	0.1623	0.1626	0.1598	0.1202	0.1641	0.1580	0.1428	0.1185	0.1177	0.1523	0.1603	0.1449
	50%	**0.1537**	0.1342	0.1330	0.1343	0.1289	0.1252	0.1324	0.1353	0.1186	0.1328	0.1216	0.1262	0.1481	0.1289
	0%	0.5547	0.5547	0.5369	0.5429	0.4853	0.3321	0.4887	0.5158	**0.5744**	0.4904	0.4385	0.5516	0.4561	0.4993
	15%	**0.4917**	0.4849	0.4656	0.4488	0.4414	0.2843	0.4525	0.4020	0.3316	0.3370	0.3405	0.3876	0.4450	0.4525
Health	30%	**0.2992**	0.2446	0.2339	0.2404	0.2313	0.1880	0.2646	0.2055	0.1970	0.1983	0.1653	0.2268	0.2526	0.2612
	40%	0.1704	0.1510	0.1478	0.1494	0.1572	0.1192	0.1507	0.1529	0.1423	0.1110	0.1061	0.1510	**0.1744**	0.1495
	50%	**0.1120**	0.0983	0.0969	0.1033	0.1036	0.1005	0.0964	0.0977	0.0954	0.0859	0.0919	0.1031	0.1084	0.1095
	0%	0.1483	0.1624	0.1751	0.1959	0.1788	0.1414	0.1668	0.1647	0.1518	0.1519	0.1481	0.1806	**0.2031**	0.1615
	15%	**0.2253**	0.1631	0.1769	0.2002	0.1885	0.1900	0.1808	0.1201	0.1580	0.1669	0.1795	0.1937	0.2012	0.1688
Image	30%	**0.2958**	0.2386	0.2504	0.2473	0.2416	0.2390	0.2200	0.2144	0.1778	0.1666	0.2076	0.2172	0.2941	0.1665
	40%	0.3436	0.3228	0.3160	0.3295	0.3236	0.2939	0.3200	0.3212	0.2906	0.2861	0.2832	**0.3459**	0.3436	0.3097
	50%	0.4106	0.3771	0.3796	0.3942	0.3830	0.3942	0.3680	0.3845	0.3623	0.3816	0.3820	0.4237	**0.4241**	0.3780
	0%	**0.1366**	0.1345	0.1344	0.1237	0.0896	0.0664	0.1224	0.1162	0.0315	0.0697	0.0426	0.1215	0.0883	0.0303
	15%	0.0751	0.0625	0.0639	**0.0812**	0.0671	0.0365	0.0674	0.0119	0.0449	0.0378	0.0512	0.0734	0.0724	0.0663
Langua	30%	0.0414	0.0420	0.0395	0.0489	0.0371	0.0430	0.0427	0.0348	0.0321	0.0428	0.0312	0.0454	**0.0538**	0.0466
	40%	**0.0441**	0.0393	0.0363	0.0391	0.0401	0.0388	0.0413	0.0374	0.0365	0.0372	0.0331	0.0379	0.0432	0.0374
	50%	0.0330	0.0324	0.0317	0.0322	0.0313	0.0334	0.0334	0.0319	0.0306	0.0308	0.0268	0.0340	**0.0342**	0.0313
	0%	0.7568	0.7769	**0.7813**	0.7781	0.6667	0.7262	0.7408	0.7410	0.4663	0.3146	0.2517	0.7790	0.7260	0.7179
	15%	**0.5955**	0.5884	0.5547	0.5659	0.5016	0.4647	0.5104	0.4774	0.3276	0.1707	0.0115	0.5014	0.5246	0.5216
Medica	30%	**0.2339**	0.1802	0.1819	0.2294	0.0997	0.1580	0.1803	0.1207	0.0985	0.1055	0.0496	0.1457	0.2092	0.2210
	40%	0.1084	0.1026	0.1032	**0.1101**	0.0912	0.0701	0.1034	0.1046	0.0835	0.1020	0.0624	0.1013	0.1040	0.1070
	50%	0.0619	0.0628	0.0611	0.0661	0.0651	0.0651	0.0588	0.0614	0.0595	0.0509	0.0417	0.0691	**0.0739**	0.0669
	0%	**0.3462**	0.3426	0.3326	0.3449	0.2570	0.2131	0.2158	0.2689	0.3371	0.2760	0.1432	0.3374	0.2025	0.2732
	15%	**0.2876**	0.2799	0.2784	0.2448	0.2293	0.1055	0.2334	0.2091	0.1053	0.1305	0.1371	0.2424	0.2060	0.2185
Recrea	30%	**0.1973**	0.1749	0.1666	0.1728	0.1725	0.1050	0.1790	0.1664	0.1298	0.1464	0.1273	0.1825	0.1723	0.1835
	40%	**0.1518**	0.1388	0.1413	0.1416	0.1298	0.1039	0.1398	0.1367	0.1173	0.1095	0.1183	0.1413	0.1442	0.1457
	50%	0.1238	0.1179	0.1192	0.1184	0.1183	0.1198	0.1193	0.1170	0.1126	0.1161	0.1134	0.1151	**0.1291**	0.1136
	0%	0.4930	0.4851	0.4795	0.4791	0.4463	0.3372	0.4371	0.4563	0.4838	**0.5132**	0.2789	0.4773	0.4090	0.4536
	15%	**0.4084**	0.4062	0.4000	0.4076	0.3690	0.2065	0.3781	0.3355	0.3001	0.2704	0.2565	0.3997	0.3739	0.3945
Refere	30%	**0.2134**	0.1822	0.1662	0.1912	0.1735	0.1740	0.1760	0.1638	0.1402	0.1804	0.1623	0.2031	0.1989	0.1846
	40%	0.1113	0.1118	0.1155	0.1112	0.1084	0.1144	0.1122	0.1153	0.0853	0.1266	0.0952	0.1103	**0.1338**	0.0957
	50%	0.0779	0.0743	0.0624	0.0729	0.0763	0.0528	0.0676	0.0632	0.0649	0.0594	0.0698	0.0702	**0.0808**	0.0782
	0%	0.6540	0.6320	0.6420	0.6718	0.6755	0.6127	0.5766	0.6412	0.6789	0.5641	0.5832	0.6299	**0.6876**	0.6010
	15%	0.6159	0.5776	0.6352	0.2540	0.6186	0.5880	0.4966	0.5422	**0.6549**	0.5184	0.5057	0.5866	0.6214	0.4546
Scene	30%	**0.4923**	0.4207	0.4671	0.4304	0.4697	0.4350	0.3906	0.3631	0.4439	0.4145	0.4170	0.4543	0.4796	0.3529
	40%	0.3672	0.3116	0.3269	0.3222	0.3749	0.3158	0.3477	0.3391	0.3584	0.3676	**0.3972**	0.3238	0.3927	0.3314
	50%	0.2747	0.2664	0.2738	0.2793	0.2614	0.2653	0.2731	0.2610	0.2550	0.2666	0.2509	0.2759	**0.2912**	0.2898
	0%	**0.3198**	0.2971	0.3033	0.2879	0.2254	0.0681	0.2185	0.2615	0.2978	0.2153	0.1422	0.3004	0.1914	0.2736
	15%	**0.2480**	0.2223	0.2176	0.1861	0.1965	0.0784	0.1989	0.1613	0.1011	0.0897	0.0870	0.1839	0.1910	0.1689
Scien	30%	**0.1384**	0.1260	0.1241	0.1190	0.1153	0.0700	0.1179	0.0990	0.0999	0.0793	0.0826	0.1198	0.1193	0.1233
	40%	**0.0985**	0.0872	0.0921	0.0870	0.0814	0.0776	0.0876	0.0884	0.0788	0.0746	0.0733	0.0877	0.0901	0.0842
	50%	**0.0763**	0.0712	0.0739	0.0728	0.0730	0.0635	0.0726	0.0727	0.0647	0.0577	0.0660	0.0726	0.0755	0.0733
	0%	0.4551	0.4612	**0.4631**	0.4545	0.2661	0.2401	0.4537	0.4471	0.1957	0.2962	0.2913	0.4018	0.3419	0.4079
	15%	**0.3822**	0.3230	0.2765	0.3416	0.2410	0.1946	0.2957	0.2433	0.1568	0.1831	0.1970	0.1599	0.2723	0.2614
Slashd	30%	0.2027	0.1520	0.1517	0.1553	0.1416	0.0790	0.1546	0.1221	0.1145	0.1156	0.0794	0.1669	0.1903	**0.2087**
	40%	0.1437	0.1362	0.1283	0.1310	0.1225	0.0956	0.1349	0.1229	0.1136	0.1053	0.1094	0.1060	**0.1512**	0.1177
	50%	0.1089	0.1074	0.1135	0.1106	0.1027	0.1051	0.1036	0.1081	0.0981	0.1040	0.0895	**0.1243**	0.1181	0.1064
	0%	**0.5930**	0.5697	0.5723	0.5680	0.5109	0.1894	0.5306	0.5485	0.5883	0.5398	0.4520	0.5884	0.5135	0.5834
	15%	**0.4950**	0.4647	0.4711	0.4847	0.4219	0.1656	0.4454	0.4118	0.3282	0.2670	0.3237	0.4426	0.4271	0.3967
Social	30%	**0.2399**	0.2058	0.1867	0.1946	0.1860	0.1360	0.2002	0.1305	0.1500	0.1393	0.1460	0.1605	0.2015	0.1934
	40%	0.1022	0.1036	0.1044	0.1072	0.1035	0.0652	0.1067	0.1069	0.0763	0.0725	0.0957	0.1029	**0.1185**	0.0977
	50%	0.0717	0.0645	0.0630	0.0670	0.0632	0.0685	0.0630	0.0650	0.0663	0.0675	0.0692	0.0635	**0.0751**	0.0638
	0%	**0.3915**	0.3836	0.3805	0.3761	0.3471	0.3004	0.3534	0.3536	0.3711	0.3625	0.2904	0.3890	0.3055	0.3633
	15%	0.3342	0.3330	0.3327	**0.3584**	0.3129	0.2876	0.3286	0.2668	0.2333	0.2888	0.2808	0.3203	0.3245	0.3056
Societ	30%	**0.2229**	0.2152	0.1988	0.2193	0.2060	0.1440	0.2116	0.1908	0.1986	0.1366	0.1474	0.2104	0.2173	0.2042
	40%	0.1514	0.1449	0.1341	0.1427	0.1585	0.1394	0.1321	0.1425	0.1345	0.1162	0.1192	0.1301	**0.1612**	0.1334
	50%	**0.1308**	0.1188	0.1216	0.1222	0.1257	0.1177	0.1213	0.1213	0.1071	0.1217	0.1206	0.1196	0.1211	0.1237
	0%	0.5750	0.6142	0.6000	**0.6147**	0.5679	0.5533	0.6120	0.6017	0.5181	0.5153	0.4776	0.5295	0.5791	0.5621
	15%	**0.5781**	0.5755	0.5711	0.5455	0.5579	0.5310	0.5454	0.5159	0.5058	0.3689	0.4470	0.5369	0.5606	0.5475
Yeast	30%	**0.5555**	0.5188	0.5100	0.5317	0.5181	0.4990	0.5212	0.4927	0.5019	0.4832	0.3909	0.5283	0.5546	0.5277
	40%	0.5139	0.4638	0.4805	0.5001	0.4767	0.4557	0.4550	0.4730	0.3972	0.2886	0.4618	0.4100	**0.5235**	0.4568
	50%	0.4873	0.4484	0.4544	0.4696	0.4256	0.4434	0.4676	0.4482	0.4087	0.3887	0.3128	0.4419	**0.5260**	0.5013

Table 4
Macro-F1 scores on real-world datasets under different label noise levels: 0% (noise-free), 15%, 30%, 40%, and 50%. The highest score in each row is shown in **bold**, and the second-highest in underlined.

Dataset	Noise	RLBMLFS	SRFS	RFSFS	LMLFS	SPLDG	DRMFS	MLFS-GL	MRDM	SSFS	SCMFS	SCLS	SCNMF	SLCLR	GNN-DE
Arts	0%	**0.1658**	0.1657	0.1633	0.1553	0.1078	0.1146	0.1597	0.1301	0.1052	0.1299	0.0534	0.1443	0.1026	0.1320
	15%	**0.1434**	0.1286	0.1397	0.1190	0.1095	0.0940	0.0666	0.0782	0.1188	0.0654	0.0703	0.1214	0.0943	0.0856
	30%	**0.1124**	0.1086	0.1085	0.1037	0.1040	0.0685	0.0945	0.0714	0.1068	0.0881	0.0873	0.0998	0.1073	0.1017
	40%	**0.1130**	0.1080	0.1060	0.1063	0.1022	0.0866	0.0987	0.0925	0.1072	0.1075	0.0940	0.1030	0.1049	0.1045
	50%	**0.1075**	0.1050	0.1055	0.1057	0.1016	0.1023	0.1019	0.1023	0.1051	0.1040	0.1005	0.1048	0.1043	0.1007
Educat	0%	0.1353	0.1423	0.1209	0.1242	0.1068	**0.1868**	0.1228	0.0880	0.0936	0.1151	0.0488	0.1277	0.1114	0.1200
	15%	**0.1021**	0.1000	0.0988	0.0991	0.0895	0.0473	0.0481	0.0457	0.0930	0.0345	0.0394	0.0725	0.0944	0.0639
	30%	**0.0814**	0.0778	0.0783	0.0713	0.0731	0.0456	0.0580	0.0382	0.0786	0.0514	0.0597	0.0634	0.0761	0.0701
	40%	**0.0760**	0.0728	0.0734	0.0732	0.0712	0.0597	0.0664	0.0586	0.0731	0.0729	0.0582	0.0714	0.0726	0.0657
	50%	0.0722	0.0723	0.0723	0.0698	0.0709	0.0656	0.0690	0.0677	0.0716	**0.0727**	0.0699	0.0694	0.0724	0.0683
Emotio	0%	**0.5343**	0.3977	0.2823	0.3006	0.1873	0.1894	0.3038	0.1677	0.3006	0.3006	0.1670	0.2119	0.1859	0.3025
	15%	**0.5103**	0.3328	0.2186	0.1929	0.2198	0.2515	0.3010	0.1677	0.2429	0.3281	0.1677	0.2458	0.3295	0.2092
	30%	0.4120	0.3743	0.3882	0.3540	0.2982	0.3192	0.1226	0.1677	0.3499	0.1296	0.2159	0.2264	**0.4336**	0.1884
	40%	**0.3628**	0.2786	0.3177	0.2966	0.2905	0.1838	0.2420	0.1315	0.3437	0.3004	0.2646	0.3075	0.3679	0.2287
	50%	0.3728	0.3467	0.3257	0.3921	0.3107	0.3271	0.2896	0.2768	0.3716	0.3693	0.3076	0.3544	**0.4571**	0.3354
Entert	0%	0.2141	0.2139	0.2083	0.2075	0.1766	0.0707	**0.2155**	0.1907	0.1736	0.1918	0.1191	0.2121	0.1770	0.1715
	15%	**0.1843**	0.1777	0.1739	0.1740	0.1571	0.0735	0.1039	0.1433	0.1708	0.1140	0.1108	0.1672	0.1594	0.1435
	30%	**0.1698**	0.1471	0.1433	0.1306	0.1305	0.0803	0.1101	0.0962	0.1406	0.1265	0.0888	0.1438	0.1411	0.1440
	40%	**0.1373**	0.1193	0.1209	0.1160	0.1204	0.0966	0.1075	0.0904	0.1196	0.1182	0.0982	0.1145	0.1166	0.1110
	50%	**0.1187**	0.1101	0.1101	0.1092	0.1080	0.1046	0.1028	0.1093	0.1102	0.1105	0.1034	0.1076	0.1127	0.1073
Health	0%	**0.2326**	0.2327	0.2259	0.2323	0.1847	0.0558	0.2283	0.1967	0.1908	0.1991	0.1321	0.2323	0.1732	0.1970
	15%	**0.1823**	0.1783	0.1710	0.1561	0.1468	0.0508	0.0925	0.0974	0.1601	0.1118	0.0759	0.1371	0.1550	0.1389
	30%	**0.1135**	0.1037	0.1019	0.1023	0.0972	0.0571	0.0812	0.0781	0.0900	0.0898	0.0663	0.0969	0.1035	0.1098
	40%	**0.0875**	0.0876	0.0853	0.0862	0.0851	0.0699	0.0768	0.0667	0.0857	0.0853	0.0677	0.0819	0.0827	0.0844
	50%	0.0773	0.0740	0.0719	0.0760	0.0747	0.0730	0.0741	0.0688	0.0730	0.0731	0.0699	0.0745	0.0768	**0.0751**
Image	0%	0.1395	0.1530	0.1617	0.1807	0.1696	0.1375	0.1422	0.1490	0.1525	0.1570	0.1432	0.1668	**0.1933**	0.1650
	15%	**0.1911**	0.1527	0.1587	0.1848	0.1741	0.1801	0.1496	0.1625	0.1718	0.1053	0.1735	0.1858	0.1907	0.1652
	30%	**0.2612**	0.2307	0.2254	0.2301	0.2215	0.2197	0.1642	0.1620	0.2034	0.1955	0.1823	0.1939	0.2535	0.1553
	40%	**0.2975**	0.2815	0.2811	0.2905	0.2813	0.2579	0.2593	0.2552	0.2752	0.2816	0.2441	0.3009	0.2907	0.2722
	50%	**0.3426**	0.3263	0.3200	0.3349	0.3282	0.3377	0.3086	0.3219	0.3208	0.3182	0.3292	0.3432	0.3521	0.3252
Langua	0%	0.0288	**0.0325**	0.0304	0.0281	0.0211	0.0162	0.0080	0.0175	0.0259	0.0259	0.0100	0.0236	0.0194	0.0250
	15%	0.0291	0.0253	0.0276	0.0284	0.0221	0.0188	0.0112	0.0176	0.0237	0.0145	0.0092	**0.0263**	0.0317	0.0122
	30%	0.0322	0.0307	0.0308	0.0318	0.0294	0.0290	0.0234	0.0275	0.0309	0.0273	0.0130	**0.0343**	0.0336	0.0282
	40%	**0.0364**	0.0339	0.0324	0.0328	0.0339	0.0329	0.0320	0.0316	0.0347	0.0323	0.0159	0.0312	0.0356	0.0274
	50%	**0.0305**	0.0297	0.0295	0.0306	0.0292	0.0304	0.0282	0.0281	0.0298	0.0302	0.0221	0.0321	0.0316	0.0279
Medica	0%	0.2768	0.2992	0.3001	0.2987	0.2185	0.2670	0.1167	0.0856	0.2618	0.2584	0.0572	**0.3021**	0.2523	0.2713
	15%	0.2313	0.2021	0.1972	0.1989	0.1598	0.1689	0.1223	0.0470	0.1612	0.1454	0.0016	0.1740	0.1595	**0.2509**
	30%	0.1270	0.1000	0.1027	0.1227	0.0669	0.0652	0.0610	0.0655	0.0966	0.0651	0.0043	0.0684	0.1017	**0.1339**
	40%	**0.0716**	0.0716	0.0732	0.0738	0.0623	0.0411	0.0583	0.0621	0.0721	0.0723	0.0081	0.0584	0.0716	0.0704
	50%	**0.0515**	0.0505	0.0501	0.0520	0.0502	0.0371	0.0493	0.0447	0.0489	0.0531	0.0236	0.0488	0.0553	0.0535
Recrea	0%	**0.2145**	0.2110	0.2044	0.2065	0.1569	0.1079	0.1756	0.1691	0.1317	0.1576	0.0802	0.2123	0.1258	0.1993
	15%	**0.1732**	0.1713	0.0904	0.1646	0.1345	0.0565	0.0616	0.0780	0.1371	0.0998	0.0748	0.1403	0.1214	0.1184
	30%	**0.1318**	0.1217	0.1174	0.1207	0.1248	0.0769	0.0912	0.1032	0.1193	0.1160	0.0915	0.1249	0.1168	0.1244
	40%	**0.1233**	0.1183	0.1175	0.1180	0.1128	0.0866	0.1006	0.0941	0.1177	0.1173	0.0990	0.1198	0.1148	0.1157
	50%	**0.1090**	0.1089	0.1084	0.1084	0.1073	0.1046	0.1049	0.1075	0.1088	0.1094	0.1039	0.1081	0.1128	0.1048
Refere	0%	**0.1300**	0.1247	0.1233	0.1227	0.0958	0.0296	0.1203	0.1093	0.0940	0.1136	0.0690	0.1167	0.0862	0.1011
	15%	**0.1000**	0.0969	0.0935	0.0905	0.0830	0.0373	0.0547	0.0391	0.0876	0.0346	0.0449	0.0912	0.0778	0.0716
	30%	**0.0753**	0.0732	0.0731	0.0725	0.0688	0.0486	0.0578	0.0527	0.0724	0.0619	0.0579	0.0702	0.0674	0.0674
	40%	**0.0614**	0.0617	0.0633	0.0618	0.0611	0.0573	0.0530	0.0563	0.0625	0.0617	0.0532	0.0579	0.0618	0.0560
	50%	0.0568	0.0545	0.0519	0.0554	0.0564	0.0497	0.0533	0.0494	0.0544	0.0522	0.0536	0.0543	0.0566	**0.0567**
Scene	0%	0.6558	0.5676	0.5769	0.6005	0.6062	0.5541	**0.6823**	0.5044	0.5719	0.6446	0.5880	0.5684	0.6258	0.6146
	15%	0.6251	**0.6395**	0.5244	0.5291	0.5594	0.5320	0.5853	0.4752	0.4556	0.4813	0.4636	0.5358	0.5613	0.0640
	30%	**0.4537**	0.4470	0.4203	0.3939	0.4302	0.4034	0.4115	0.3743	0.3652	0.3325	0.3914	0.4139	0.4470	0.3476
	40%	0.3356	0.2864	0.3003	0.2951	0.3468	0.2896	0.3386	0.3436	0.3168	0.3118	0.3625	0.3069	**0.3621**	0.2993
	50%	**0.2564**	0.2482	0.2558	0.2634	0.2450	0.2438	0.2376	0.2527	0.2543	0.2527	0.2311	0.2566	0.2667	0.2644
Scien	0%	0.1319	0.1304	**0.1380**	0.1170	0.0952	0.0176	0.1090	0.0915	0.0864	0.0947	0.0531	0.1107	0.1380	0.1159
	15%	**0.0940**	0.0891	0.0875	0.0799	0.0763	0.0234	0.0333	0.0353	0.0792	0.0237	0.0225	0.0657	0.0891	0.0624
	30%	**0.0770**	0.0735	0.0730	0.0686	0.0669	0.0468	0.0589	0.0445	0.0704	0.0606	0.0472	0.0715	0.0668	0.0701
	40%	**0.0694**	0.0666	0.0671	0.0644	0.0618	0.0495	0.0598	0.0517	0.0670	0.0671	0.0526	0.0642	0.0654	0.0628
	50%	**0.0620**	0.0600	0.0605	0.0615	0.0611	0.0560	0.0587	0.0566	0.0600	0.0610	0.0584	0.0616	0.0618	0.0605
Slashd	0%	0.2374	0.2614	**0.2671**	0.2516	0.1324	0.0964	0.0740	0.1355	0.2546	0.2517	0.1369	0.2306	0.2671	0.2560
	15%	**0.2053**	0.1768	0.1274	0.1294	0.1144	0.0802	0.0814	0.0756	0.1683	0.1006	0.1037	0.0829	0.1371	0.0781
	30%	**0.1271**	0.1071	0.1098	0.1001	0.0971	0.0678	0.0841	0.0773	0.1131	0.0798	0.0849	0.0960	0.1066	0.0768
	40%	**0.1000**	0.1030	0.1059	0.0924	0.0947	0.0722	0.0878	0.0788	0.1042	0.0971	0.0787	0.0797	0.1079	0.0774
	50%	0.0960	0.0933	0.0925	0.0950	0.0922	0.0916	0.0899	0.0937	0.0919	0.0903	0.0827	0.0967	0.0967	**0.0937**
Social	0%	**0.1671**	0.1671	0.1635	0.1559	0.1182	0.0291	0.1567	0.1177	0.1349	0.1362	0.0937	0.1657	0.1027	0.1391
	15%	**0.1232**	0.1150	0.1098	0.1017	0.0834	0.0289	0.0537	0.0372	0.0910	0.0494	0.0450	0.0901	0.0878	0.0787
	30%	**0.0891**	0.0659	0.0681	0.0649	0.0652	0.0409	0.0495	0.0425	0.0686	0.0513	0.0437	0.0556	0.0662	0.0590
	40%	**0.0564**	0.0580	0.0580	0.0584	0.0553	0.0429	0.0464	0.0402	0.0592	0.0585	0.0468	0.0560	0.0575	0.0545
	50%	**0.0505**	0.0470	0.0456	0.0488	0.0487	0.0475	0.0482	0.0492	0.0452	0.0489	0.0502	0.0487	0.0497	0.0498
Societ	0%	**0.1312**	0.1243	0.1259	0.1227	0.1021	0.0413	0.1195	0.1050	0.1024	0.1114	0.0415	0.1273	0.1015	0.1117
	15%	**0.1085**	0.1070	0.1040	0.0946	0.0961	0.0551	0.0585	0.0656	0.1033	0.0448	0.0531	0.0897	0.0988	0.0860
	30%	**0.1028**	0.1000	0.1012	0.0979	0.0947	0.0705	0.0835	0.0692	0.0998	0.0945	0.0802	0.0971	0.1012	0.1009
	40%	**0.0988**	0.0945	0.0934	0.0938	0.0941	0.0887	0.0910	0.0804	0.0938	0.0939	0.0857	0.0921	0.0978	0.0918
	50%	**0.1023**	0.0969	0.0972	0.0988	0.0968	0.0914	0.0917	0.0973	0.0977	0.0978	0.0958	0.0979	0.0985	0.0981
Yeast	0%	0.3214	**0.3637**	0.3578	0.3636	0.2667	0.2397	0.2097	0.1774	0.3636	0.3637	0.1544	0.2028	0.2771	0.2958
	15%	0.3598	**0.3800**	0.3494	0.2537	0.2676	0.2297	0.1851	0.1687	0.2680	0.3098	0.1765	0.2224	0.2659	0.2409
	30%	**0.3356**	0.2777	0.2645	0.2981	0.2794	0.2674	0.3094	0.2769	0.2982	0.2675	0.2841	0.2640	0.3247	0.2900
	40%	0.3548	0.2762	0.3018	0.3072	0.3308	0.2786	0.2561	0.1818	0.3115	0.2813	0.2792	0.2040	**0.3695**	0.2531
	50%	0.3548	0.3172	0.3105	0.3283	0.3073	0.3127	0.3222	0.2805	0.3187	0.3165	0.2462	0.2853	**0.3649**	0.3580

Table 5
Hamming loss scores on real-world datasets under different label noise levels. The lowest score is shown in **bold**, while the second-lowest score is underlined.

Dataset	Noise	RLBMLFS	SRFS	RFSFS	LMLFS	SPLDG	DRMFS	SSFS	SCMFS	MLFS-GL	MRDM	SCLS	SCNMF	SLCLR	GNN-DE
Arts	0%	**0.0566**	0.0575	0.0579	0.0585	0.0603	0.0603	0.0615	0.0588	0.0568	0.0597	0.0631	0.0599	0.0623	0.0591
	15%	**0.0670**	0.0674	0.0695	0.0705	0.0700	0.0712	0.0722	0.1117	0.0781	0.0724	0.0902	0.0699	0.0699	0.0690
	30%	**0.1367**	0.1535	0.1586	0.1589	0.1497	0.1391	0.1564	0.1715	0.1767	0.1395	0.1566	0.1453	0.1475	0.1399
	40%	0.2327	0.3041	0.2857	0.2951	0.2295	0.2632	0.2946	0.2992	0.3042	0.2600	0.3139	0.2937	**0.2157**	0.2869
	50%	0.4421	0.4690	0.4737	0.4615	0.4625	0.4796	0.4864	0.4501	0.5051	0.4883	0.5061	0.4809	0.4427	**0.4341**
Educat	0%	**0.0388**	0.0394	0.0393	0.0405	0.0421	0.0450	0.0414	0.0410	0.0392	0.0436	0.0443	0.0414	0.0415	0.0400
	15%	0.0495	**0.0482**	0.0490	0.0519	0.0518	0.0526	0.0519	0.0902	0.0593	0.0591	0.0622	0.0513	0.0503	0.0521
	30%	0.1229	0.1421	0.1478	0.1349	0.1365	0.1132	0.1452	0.1332	0.1450	**0.1109**	0.1560	0.1260	0.1278	0.1184
	40%	**0.2064**	0.2706	0.2849	0.2731	0.2069	0.2919	0.2782	0.2817	0.2910	0.2183	0.2871	0.2726	0.1914	0.2701
	50%	**0.4487**	0.4748	0.4752	0.4877	0.4612	0.4452	0.4690	0.4795	0.5041	0.5016	0.5325	0.4751	0.4553	0.5148
Emotio	0%	**0.2292**	0.2857	0.3176	0.3094	0.3143	0.3411	0.3094	0.3094	0.3094	0.3994	0.3994	0.3080	0.2897	0.3071
	15%	**0.2641**	0.3478	0.3326	0.3113	0.3045	0.4126	0.3317	0.3242	0.3094	0.3994	0.3994	0.3291	0.2890	0.3214
	30%	0.3618	0.4217	0.4142	0.3139	0.3390	0.4808	0.3865	0.3748	0.4093	0.3994	0.5098	0.3643	**0.3181**	0.3214
	40%	0.3305	0.3502	0.3610	0.3354	0.3734	0.3594	0.4001	0.3835	0.3647	0.3917	0.4297	0.5127	0.3270	**0.4515**
	50%	**0.3352**	0.3368	0.3769	0.3373	0.3706	0.3603	0.4027	0.3938	0.4451	0.6020	0.5485	0.3214	0.3214	0.3896
Entert	0%	**0.0551**	0.0569	0.0568	0.0576	0.0624	0.0688	0.0615	0.0594	0.0555	0.0613	0.0634	0.0586	0.0627	0.0620
	15%	**0.0671**	0.0671	0.0704	0.0725	0.0731	0.0780	0.0740	0.1101	0.0804	0.0706	0.0839	0.0700	0.0736	0.0731
	30%	0.1384	0.1454	0.1477	0.1449	0.1528	0.1394	0.1498	0.1765	0.1655	0.1444	0.1437	0.1473	0.1570	**0.1362**
	40%	**0.2047**	0.3034	0.2892	0.2936	0.2233	0.3215	0.2918	0.2875	0.3106	0.2619	0.3318	0.2742	0.2080	0.3046
	50%	**0.4357**	0.4779	0.4881	0.4590	0.4608	0.4367	0.4665	0.4693	0.5014	0.5201	0.5372	0.4711	0.4400	0.4726
Health	0%	**0.0381**	0.0386	0.0387	0.0391	0.0428	0.0500	0.0427	0.0407	0.0382	0.0423	0.0461	0.0406	0.0433	0.0395
	15%	**0.0470**	0.0490	0.0490	0.0519	0.0530	0.0574	0.0496	0.0904	0.0601	0.0585	0.0676	0.0545	0.0503	0.0504
	30%	0.1236	0.1369	0.1397	0.1306	0.1363	**0.1113**	0.1320	0.1575	0.1444	0.1271	0.1419	0.1271	0.1332	0.1165
	40%	0.2464	0.2760	0.2778	0.2720	0.2165	0.2710	0.2757	0.2795	0.2762	0.2646	0.3033	0.2762	**0.2033**	0.2745
	50%	0.4606	0.4900	0.4910	0.4741	0.4850	0.4711	0.4958	0.4868	0.5110	0.5428	0.5101	0.4790	0.4619	**0.4478**
Image	0%	0.3490	0.3513	0.3570	0.3290	0.3343	0.3622	0.3544	0.3535	0.3579	**0.3210**	0.3453	0.3443	0.3243	0.3320
	15%	0.3559	0.3566	0.3708	0.3299	0.3445	0.3717	0.3512	0.3883	0.3392	**0.3245**	0.3405	0.3470	0.3288	0.3353
	30%	0.3582	0.3852	0.3822	0.3606	0.3762	0.3928	0.3614	0.4420	0.3970	0.3603	0.3723	0.3965	0.3665	**0.3338**
	40%	**0.4071**	0.4166	0.4146	0.4242	0.4168	0.4280	0.4100	0.3942	0.4635	0.4430	0.4405	0.3973	0.3943	0.4168
	50%	0.4330	0.4685	0.4685	0.4685	0.4627	0.4799	0.4673	0.4626	0.4785	0.4238	0.4660	0.4038	0.4168	**0.4448**
Langua	0%	0.0170	0.0169	0.0165	**0.0157**	0.0158	0.0180	0.0157	0.0158	0.0159	0.0165	0.0278	0.0934	0.0157	0.0158
	15%	0.0345	0.0469	0.0442	0.0231	0.0247	0.0712	0.0261	0.0636	**0.0158**	0.0407	0.1102	0.0252	0.0222	0.0255
	30%	0.1152	0.1026	0.1321	0.0868	0.0922	**0.0830**	0.0979	0.1137	0.1058	0.0859	0.1945	0.0934	0.1014	0.0862
	40%	**0.1774**	0.2293	0.2530	0.2375	0.2156	0.2395	0.2190	0.2468	0.2384	0.2321	0.2792	0.2052	0.2023	0.2369
	50%	**0.4854**	0.4948	0.4880	0.4986	0.4939	0.4913	0.4944	0.4963	0.5190	0.5052	0.5440	0.4854	0.4714	0.5181
Medica	0%	0.0120	0.0114	**0.0112**	0.0113	0.0159	0.0135	0.0127	0.0127	0.0238	0.0339	0.0253	0.0183	0.0131	0.0137
	15%	0.0229	0.0235	0.0250	0.0254	0.0273	0.0273	0.0283	0.0649	0.0253	0.0657	0.1369	0.0271	0.0259	**0.0232**
	30%	0.0804	0.1115	0.1089	0.0996	0.0973	0.1273	0.1092	0.1218	0.1501	0.1012	0.0893	0.1086	**0.0754**	0.0932
	40%	0.1412	0.2446	0.2430	0.2483	**0.0273**	0.0273	0.2544	0.2516	0.2497	0.2211	0.1298	0.1707	0.0523	0.2376
	50%	0.4734	0.4824	0.4861	0.4693	0.4645	0.6073	0.4950	0.4745	0.4948	0.4977	0.5799	0.4602	**0.4508**	0.4773
Recrea	0%	0.0576	0.0590	0.0588	0.0586	0.0612	0.0610	0.0617	0.0592	**0.0571**	0.0590	0.0644	0.0588	0.0619	0.0596
	15%	0.0693	0.0727	0.0722	0.0740	0.0743	0.0756	0.0746	0.1107	0.0799	0.0860	0.0892	0.0717	**0.0690**	0.0702
	30%	**0.1396**	0.1554	0.1579	0.1527	0.1578	0.1471	0.1522	0.1721	0.1623	0.1588	0.1678	0.1465	0.1542	0.1419
	40%	0.2522	0.2894	0.2937	0.2844	0.2225	0.2665	0.2844	0.2951	0.3004	0.2609	0.3247	0.2736	**0.2027**	0.2726
	50%	0.4461	0.4884	0.4697	0.4757	0.4553	0.4574	0.4707	0.4782	0.4736	0.5053	0.4757	0.4817	**0.4203**	0.4535
Refere	0%	**0.0270**	0.0285	0.0281	0.0282	0.0304	0.0347	0.0296	0.0287	0.0275	0.0291	0.0323	0.0276	0.0301	0.0281
	15%	0.0357	0.0392	0.0364	0.0371	0.0391	0.0450	0.0367	0.0759	0.0434	0.0435	0.0409	0.0359	0.0372	**0.0352**
	30%	0.1120	0.1320	0.1442	0.1143	0.1256	0.1231	0.1313	0.1346	0.1331	0.1218	0.1438	**0.1056**	0.1209	0.1106
	40%	0.2473	0.2723	0.2756	0.2769	0.1931	0.2791	0.2644	0.2767	0.2752	0.2272	0.2808	0.2684	**0.1908**	0.2952
	50%	0.4654	0.4906	0.4944	0.4807	0.4830	0.4781	0.4923	0.4944	0.5121	0.5124	0.4567	0.4860	0.4604	**0.4439**
Scene	0%	0.1138	0.1186	0.1169	0.1101	0.1088	0.1303	0.1333	0.1155	**0.1074**	0.1509	0.1471	0.1197	0.1074	0.1105
	15%	0.1308	0.1408	0.1212	0.1402	0.1275	0.1388	0.1622	0.1728	**0.1160**	0.1587	0.1663	0.1365	0.1227	0.1646
	30%	**0.1953**	0.2218	0.2150	0.2400	0.2158	0.2212	0.2394	0.2510	0.2378	0.2427	0.2543	0.2221	0.2190	0.2635
	40%	**0.3098**	0.3518	0.3411	0.3269	0.3214	0.3262	0.3328	0.3243	0.3572	0.3285	0.3638	0.3075	0.2952	0.3321
	50%	0.4461	0.4504	0.4628	0.4634	0.4674	0.5053	0.4740	0.4701	0.5265	0.5757	0.4532	**0.4227**	0.4418	0.5019
Scien	0%	**0.0333**	0.0338	0.0336	0.0341	0.0351	0.0371	0.0349	0.0342	0.0336	0.0346	0.0357	0.0335	0.0353	0.0348
	15%	**0.0407**	0.0438	0.0421	0.0432	0.0440	0.0470	0.0447	0.0848	0.0470	0.0487	0.0467	0.0454	0.0413	0.0418
	30%	0.1171	0.1265	0.1305	0.1229	0.1288	0.1191	0.1264	0.1440	0.1262	0.1203	0.1444	0.1342	0.1202	**0.1149**
	40%	**0.2085**	0.2637	0.2761	0.2717	0.2078	0.2916	0.2735	0.2677	0.2737	0.2258	0.2985	0.2670	0.1979	0.2674
	50%	**0.4501**	0.4803	0.4714	0.4772	0.4740	0.5026	0.4817	0.4826	0.4949	0.5098	0.5122	0.4713	0.4600	0.4716
Slashd	0%	**0.0442**	0.0450	0.0447	0.0456	0.0508	0.0749	0.0466	0.0465	0.0545	0.0611	0.0620	0.0449	0.0469	0.0492
	15%	0.0563	0.0602	0.0637	0.0744	0.0718	0.0988	0.0741	0.1060	0.0571	0.1365	0.0780	0.0674	0.0580	**0.0495**
	30%	**0.1179**	0.1731	0.1828	0.1502	0.1442	0.2825	0.1631	0.2016	0.1849	0.1236	0.2774	0.1212	0.1326	0.1292
	40%	**0.1755**	0.1994	0.2901	0.2636	0.1691	0.3471	0.2192	0.2852	0.2248	0.1803	0.2662	0.1896	0.1601	0.1845
	50%	0.4298	0.4203	0.4352	0.3744	0.4793	0.6016	0.4846	0.4159	0.5266	0.5566	0.5815	**0.3907**	0.4284	0.4558
Social	0%	**0.0216**	0.0222	0.0224	0.0228	0.0248	0.0327	0.0243	0.0236	0.0221	0.0242	0.0275	0.0225	0.0245	0.0238
	15%	**0.0259**	0.0312	0.0310	0.0326	0.0342	0.0415	0.0340	0.0720	0.0415	0.0436	0.0408	0.0354	0.0316	0.0341
	30%	0.1028	0.1071	0.1121	0.1126	0.1112	0.1092	0.1084	0.1454	0.1282	0.1099	0.1322	0.1207	0.1179	**0.1006**
	40%	0.2608	0.2677	0.2698	0.2643	0.1991	0.3175	0.2621	0.2595	0.2700	**0.1968**	0.2845	0.2641	0.1845	0.2662
	50%	0.4685	0.4712	0.4676	0.4796	0.4780	0.4600	0.4842	0.4777	0.4938	0.4967	0.5040	0.4666	**0.4491**	0.4678
Societ	0%	**0.0542**	0.0548	0.0549	0.0552	0.0554	0.0606	0.0563	0.0553	0.0545	0.0554	0.0590	0.0558	0.0570	0.0561
	15%	**0.0637**	0.0641	0.0650	0.0665	0.0679	0.0686	0.0659	0.1053	0.0712	0.0723	0.0701	0.0672	0.0635	0.0660
	30%	**0.1327**	0.1511	0.1655	0.1471	0.1509	0.1330	0.1579	0.1699	0.1541	0.1468	0.1808	0.1488	0.1552	0.1450
	40%	0.2643	0.2876	0.2658	0.2874	0.2435	0.2806	0.2932	0.2835	0.3040	0.2593	0.2979	0.2695	**0.2200**	0.2633
	50%	**0.4614**	0.4789	0.4877	0.4744	0.4703	0.4799	0.4730	0.4812	0.4760	0.5159	0.4924	0.4558	0.4576	0.4827
Yeast	0%	0.2325	**0.2153**	0.2206	0.2153	0.2412	0.2369	0.2153	0.2153	0.2720	0.2560	0.2419	0.2475	0.2297	0.2554
	15%	**0.2433**	0.2457	0.2453	0.2528	0.2575	0.2638	0.2525	0.2785	0.2560	0.3053	0.2841	0.2454	0.2479	0.2540
	30%	**0.2685**	0.2952	0.2928	0.2745	0.2870	0.2778	0.3129	0.3189	0.3711	0.3020	0.4138	0.2782	0.2950	0.2639
	40%	**0.3092**	0.3355	0.3307	0.3361	0.3376	0.3153	0.3632	0.3490	0.4373	0.4491	0.4009	0.3247	0.2981	0.3742
	50%	0.3849	0.4151	0.3906	0.4049	0.4526	0.4174	0.4423	0.4086	0.5223	0.5088	0.5203	0.3593	**0.3450**	0.4150

Table 6
Zero-one loss scores on real-world datasets. The lowest score is **bold**, and the second-lowest is underlined. In cases of ties, priority is given to RLBMLFS.

Dataset	Noise	RLBMLFS	SRFS	RFSFS	LMLFS	SPLDG	DRMFS	MLFS-GL	MRDM	SSFS	SCMFS	SCLS	SCNMF	SLCLR	GNN-DE
Arts	0%	0.8380	**0.7879**	0.7883	0.8831	0.8560	0.8613	0.8978	0.8700	0.7986	0.8020	1.0000	0.7950	0.8757	0.8197
	15%	**0.8520**	0.8550	0.8710	0.8910	0.8760	0.9010	0.9360	0.9310	0.8850	0.9180	0.9310	0.9000	0.8970	0.8997
	30%	**0.9803**	0.9884	0.9923	0.9920	0.9872	0.9943	0.9957	0.9937	0.9912	0.9941	0.9940	0.9900	0.9850	0.9876
	40%	0.9753	1.0000	0.9993	0.9996	0.9861	0.9997	0.9998	1.0000	0.9998	0.9998	1.0000	1.0000	**0.9567**	1.0000
	50%	**1.0000**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Educat	0%	**0.7617**	0.7707	0.7787	0.7872	0.8350	0.9173	0.9186	0.7920	0.7899	0.7910	1.0000	0.7747	0.8317	0.7883
	15%	**0.8470**	0.8520	0.8670	0.8700	0.8600	0.9450	0.9290	0.9470	0.8780	0.8990	0.9530	0.9000	0.8727	0.9090
	30%	**0.9830**	0.9943	0.9943	0.9936	0.9930	0.9950	0.9958	0.9970	0.9928	0.9968	0.9997	0.9853	0.9893	0.9903
	40%	0.9590	0.9999	0.9999	0.9999	**0.9423**	1.0000	1.0000	1.0000	0.9999	0.9999	1.0000	1.0000	0.9460	0.9997
	50%	**1.0000**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Emotio	0%	0.8397	0.8579	**0.8101**	0.9156	0.8368	0.9283	0.9156	0.8481	0.9156	0.9156	0.8481	0.9241	0.8228	0.8227
	15%	**0.8020**	0.8060	0.8640	0.9160	0.8230	0.9030	0.9160	0.8480	0.8970	0.9240	0.8480	0.8780	0.8228	0.8481
	30%	**0.8402**	0.8804	0.8903	0.8467	0.8509	0.9262	0.9283	0.8481	0.9142	0.9395	1.0000	0.9030	0.8523	0.8481
	40%	0.8650	0.8959	0.9212	0.9620	0.9241	0.8931	0.9620	1.0000	0.9466	0.9269	1.0000	0.9831	**0.8228**	0.8228
	50%	0.8734	0.9156	0.9409	0.9620	0.9395	0.9128	0.9620	1.0000	0.9944	0.9578	1.0000	0.9409	0.8734	**0.9916**
Entert	0%	0.6914	0.6823	0.6851	0.7094	0.7786	0.9307	0.8318	0.7920	0.6923	0.7920	1.0000	**0.6767**	0.7954	0.7020
	15%	**0.7560**	0.7830	0.7910	0.8080	0.8160	0.9230	0.8690	0.8590	0.8190	0.8240	0.8820	0.8190	0.8347	0.8480
	30%	0.9640	0.9596	0.9646	0.9580	0.9653	0.9847	0.9740	0.9727	0.9649	0.9822	0.9773	0.9587	**0.9573**	0.9590
	40%	0.9790	0.9990	0.9989	1.0000	0.9930	0.9997	1.0000	0.9996	0.9986	0.9979	1.0000	0.9867	**0.9613**	0.9990
	50%	**1.0000**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9999	1.0000	1.0000	1.0000	1.0000
Health	0%	0.6650	0.6074	0.6117	0.6188	0.6699	0.7810	0.7666	0.5924	**0.5886**	0.6597	1.0000	0.6140	0.7176	0.6596
	15%	**0.7320**	0.7420	0.7550	0.7830	0.7750	0.8320	0.8510	0.8270	0.7620	0.7830	0.8800	0.8290	0.7957	0.7593
	30%	0.9868	0.9898	0.9911	0.9886	0.9890	0.9721	0.9928	**0.9320**	0.9886	0.9972	0.9977	0.9877	0.9250	0.9823
	40%	0.9993	0.9999	0.9998	0.9996	0.9323	1.0000	1.0000	0.9890	0.9998	0.9999	1.0000	1.0000	**0.9317**	0.9997
	50%	**1.0000**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Image	0%	0.9429	0.9350	0.9341	0.9066	0.9170	0.9460	0.9340	0.9321	0.9340	0.9338	0.9460	0.9175	**0.8988**	0.9125
	15%	**0.9100**	0.9420	0.9480	0.9610	0.9300	0.9500	0.9408	0.9310	0.9410	0.9750	0.9530	0.9390	0.9288	0.9275
	30%	**0.9275**	0.9433	0.9458	0.9296	0.9404	0.9338	0.9583	0.9588	0.9450	0.9550	0.9475	0.9450	0.9496	0.9363
	40%	0.9358	0.9367	0.9317	0.9592	0.9437	0.9417	0.9592	0.9487	0.9346	0.9367	0.9513	0.9313	**0.9275**	0.9338
	50%	0.9408	0.9567	0.9567	0.9633	0.9575	0.9613	0.9633	0.9312	0.9596	0.9563	0.9725	0.9388	0.9388	**0.9263**
Langua	0%	**0.7965**	0.8061	0.8118	0.8061	0.8142	0.8336	0.8113	0.8422	0.8070	0.8110	0.8422	0.8165	0.8285	0.8099
	15%	0.9110	0.9470	0.9430	**0.8340**	0.8380	0.9980	0.8370	0.8630	0.8930	0.8790	1.0000	0.8390	0.8370	0.8456
	30%	0.9991	0.9994	0.9994	0.9960	0.9994	0.8576	1.0000	0.8679	0.9994	1.0000	1.0000	0.8576	**0.8559**	0.8679
	40%	**0.9525**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	50%	**1.0000**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Medica	0%	0.4260	0.3560	0.3550	0.3510	0.5190	0.4250	0.7220	0.8210	0.4050	0.4110	0.8620	**0.3325**	0.4373	0.3910
	15%	0.6550	**0.6440**	0.6740	0.6790	0.7370	0.7840	0.8170	0.9590	0.7170	0.7320	1.0000	0.7030	0.7033	0.6498
	30%	**0.9769**	0.9829	0.9923	0.9855	0.9991	1.0000	0.9991	0.9974	0.9932	0.9991	1.0000	0.9974	0.9949	0.9800
	40%	0.9446	1.0000	1.0000	1.0000	0.9753	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.9974	**0.9412**	0.9520
	50%	**1.0000**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Recrea	0%	**0.7500**	0.7570	0.7570	0.7520	0.8260	0.8650	0.9190	0.8360	0.7610	0.7610	1.0000	0.7577	0.8766	0.8423
	15%	**0.8170**	0.8310	0.8350	0.8450	0.8650	0.9560	0.9510	0.9490	0.8690	0.8800	0.9400	0.8610	0.8797	0.8607
	30%	**0.9596**	0.9785	0.9764	0.9743	0.9830	0.9927	0.9883	0.9757	0.9773	0.9821	0.9877	0.9753	0.9710	0.9637
	40%	0.9870	0.9997	0.9997	0.9994	0.9883	0.9998	0.9994	0.9996	0.9993	0.9997	1.0000	0.9993	**0.9657**	0.9983
	50%	**1.0000**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Refere	0%	**0.6180**	0.6260	0.6190	0.6320	0.6630	0.7620	0.7260	0.7250	0.6300	0.6300	1.0000	0.6410	0.7073	0.6257
	15%	0.7250	0.7350	0.7330	**0.7190**	0.7570	0.9050	0.8200	0.8360	0.7500	0.7710	0.8600	0.7320	0.7480	0.7260
	30%	0.9788	0.9888	0.9918	0.9806	0.9891	0.9685	0.9947	**0.9090**	0.9892	0.9930	0.9927	0.9800	0.8937	0.9810
	40%	0.9996	0.9999	0.9999	0.9999	0.9294	1.0000	1.0000	0.9996	1.0000	1.0000	1.0000	1.0000	**0.8970**	0.9997
	50%	**1.0000**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Scene	0%	0.4820	0.4650	0.4640	0.4370	0.4430	0.4750	0.4520	0.5640	0.4750	0.4820	0.5850	0.4740	**0.4231**	0.4512
	15%	0.5330	0.5560	0.4930	0.5770	0.5200	0.5410	**0.4630**	0.6230	0.6340	0.5500	0.6480	0.5540	0.5094	0.6715
	30%	**0.7323**	0.7699	0.7578	0.8080	0.7429	0.7812	0.7883	0.8098	0.7997	0.8191	0.8337	0.7703	0.7429	0.8565
	40%	0.8950	0.9286	0.9293	0.9238	0.8999	0.9269	0.9238	0.9054	0.9099	0.9064	0.9397	0.9044	**0.8711**	0.9148
	50%	**0.9698**	0.9771	0.9792	0.9920	0.9764	0.9889	0.9920	0.9937	0.9802	0.9820	0.9782	0.9730	0.9740	0.9896
Scien	0%	**0.8010**	0.8140	0.8060	0.8160	0.8650	0.9660	0.9150	0.8710	0.8170	0.8090	1.0000	0.8033	0.8970	0.8400
	15%	**0.8740**	0.8940	0.9010	0.9180	0.9110	0.9670	0.9540	0.9710	0.9180	0.9490	0.9640	0.9260	0.9243	0.9267
	30%	**0.9915**	0.9955	0.9967	0.9944	0.9971	0.9990	0.9976	0.9987	0.9957	0.9977	0.9993	0.9973	0.9957	0.9927
	40%	0.9750	1.0000	1.0000	1.0000	0.9930	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	**0.9680**	1.0000
	50%	**0.9860**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Slashd	0%	0.6810	0.6620	**0.6590**	0.6850	0.8190	0.8330	0.8870	0.8040	0.6770	0.6930	0.7811	0.7507	0.7665	0.6902
	15%	**0.7450**	0.8040	0.8420	0.8660	0.8510	0.9750	0.9150	0.9740	0.8780	0.8690	0.9590	0.9520	0.8353	0.8684
	30%	0.9335	0.9847	0.9903	0.9751	0.9709	0.9894	0.9951	0.9888	**0.9506**	0.9936	1.0000	0.9888	0.9512	0.8638
	40%	0.9559	0.9815	0.9989	0.9907	0.9678	1.0000	0.9998	1.0000	0.9914	0.9993	1.0000	0.9967	**0.9378**	0.9743
	50%	0.9997	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	**0.8638**
Social	0%	0.5400	0.5280	0.5290	0.5320	0.5850	0.8760	0.6830	0.5430	0.5160	0.5100	1.0000	**0.5077**	0.5907	0.5423
	15%	0.6830	0.6700	**0.6630**	0.6910	0.7200	0.9230	0.8170	0.8870	0.7050	0.7100	0.8110	0.7620	0.7120	0.7323
	30%	0.9856	0.9838	0.9890	0.9879	0.9917	0.9967	0.9963	0.9960	0.9849	0.9986	0.9983	0.9960	**0.9167**	0.9843
	40%	1.0000	1.0000	1.0000	0.9999	0.9542	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	**0.9227**	1.0000
	50%	**1.0000**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Societ	0%	0.7460	0.7250	0.7260	0.7300	0.7580	0.8020	0.8320	0.7810	0.7250	0.7270	1.0000	**0.7183**	0.7990	0.7383
	15%	0.8090	0.8130	0.8130	0.8150	0.8230	0.8250	0.8680	0.8400	0.8130	0.8560	0.8420	0.8190	**0.7830**	0.8163
	30%	**0.9733**	0.9833	0.9902	0.9779	0.9839	0.9880	0.9858	0.9887	0.9863	0.9931	0.9960	0.9757	0.9750	0.9813
	40%	0.9993	0.9997	0.9998	1.0000	0.9739	1.0000	1.0000	0.9903	0.9998	0.9998	1.0000	1.0000	**0.9600**	0.9993
	50%	**1.0000**	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
Yeast	0%	0.9190	0.8680	**0.8670**	0.8680	0.9270	0.9430	0.9600	0.9810	0.8670	0.8680	0.9855	0.9027	0.9120	0.8788
	15%	0.9230	0.9070	0.9280	0.9390	0.9450	0.9490	0.9890	0.9900	0.9440	0.9640	0.9980	0.9530	0.9348	**0.8903**
	30%	**0.9570**	0.9844	0.9810	0.9790	0.9893	0.9943	1.0000	0.9855	0.9872	0.9934	0.9990	0.9654	0.9621	0.9846
	40%	**0.9782**	0.9952	0.9938	1.0000	0.9931	0.9962	1.0000	1.0000	0.9956	0.9928	1.0000	1.0000	0.9783	1.0000
	50%	**0.9855**	0.9993	0.9972	1.0000	0.9990	0.9976	1.0000	1.0000	0.9997	1.0000	1.0000	0.9979	0.9979	0.9990

Table 7
One-Error scores on real-world datasets under label noise levels: 0% (noise-free), 15%, 30%, 40%, and 50%. The lowest score is **bold**, and the second-lowest is underlined. In cases of ties, priority is given to RLBMLFS.

Dataset	Noise	RLBMLFS	RFSFS	LMLFS	SRFS	SPLDG	DRMFS	MLFS-GL	MRDM	SSFS	SCMFS	SCLS	SCNMF	SLCLR	GNN-DE
Arts	0%	**0.5767**	0.5816	0.5890	0.5834	0.6436	0.6427	0.6979	0.6927	0.5904	0.5927	0.7763	0.5920	0.6706	0.5970
	15%	**0.6467**	0.6772	0.7058	0.6634	0.6907	0.7093	0.7722	0.7467	0.6952	0.7111	0.7467	0.6743	0.7097	0.7193
	30%	0.8363	0.8599	0.8528	0.8527	0.8550	0.8497	0.8981	0.8647	0.8561	0.8886	0.9257	0.8523	**0.8277**	0.8457
	40%	**0.8223**	0.8679	0.8599	0.8659	0.8410	0.8713	0.8726	0.8537	0.8636	0.8576	0.8560	1.0000	0.8437	0.8617
	50%	0.8363	0.8537	0.8478	0.8567	0.8624	0.8423	0.8827	0.8877	0.8444	0.8532	0.8337	1.0000	**0.8297**	0.8937
Educat	0%	0.6053	0.6133	0.6256	0.6053	0.6763	**0.5696**	0.7833	0.6837	0.6192	0.6253	0.8767	0.6187	0.6747	0.6383
	15%	**0.6778**	0.6907	0.7147	0.6850	0.6860	0.8227	0.7971	0.7947	0.7166	0.7126	0.8227	0.7403	0.6967	0.7617
	30%	**0.7881**	0.8059	0.8170	0.8020	0.8112	0.8680	0.8644	0.8807	0.8064	0.8668	0.8647	0.8273	0.8097	0.8103
	40%	**0.7970**	0.8340	0.8202	0.8291	0.8283	0.8030	0.8530	0.8777	0.8326	0.8292	0.8413	1.0000	0.8217	0.8423
	50%	0.8337	0.8398	0.8508	0.8390	0.8411	0.8533	0.8604	0.8687	0.8364	0.8371	0.8710	1.0000	**0.8327**	0.8613
Emotio	0%	**0.4120**	0.6132	0.5696	0.5007	0.5724	0.5794	0.5696	0.7046	0.5696	0.5696	0.7046	0.4581	0.5907	0.5007
	15%	**0.4551**	0.5189	0.5753	0.5386	0.6666	0.5907	0.5696	0.7046	0.5541	0.5949	0.7046	0.6329	0.5654	0.5612
	30%	0.5084	0.6160	0.5921	0.5738	0.6287	0.5359	0.7046	0.7046	0.6653	0.6990	0.7046	0.7046	**0.5021**	0.5907
	40%	0.6245	0.6540	0.6174	0.6540	0.6610	0.7046	0.6751	0.7004	0.6428	0.6596	0.7046	1.0000	**0.4937**	0.7046
	50%	**0.5640**	0.5907	0.5907	0.5907	0.6301	0.6329	0.6371	0.7046	0.6610	0.6385	0.7300	1.0000	0.5907	0.7046
Entert	0%	0.6000	0.5886	0.6154	0.5940	0.7057	0.8834	0.7712	0.7283	0.5971	0.5947	0.9847	**0.5850**	0.7232	0.5960
	15%	**0.6160**	0.6578	0.6859	0.6746	0.6930	0.8562	0.7748	0.7770	0.6948	0.6936	0.7783	0.6930	0.7160	0.7453
	30%	**0.7156**	0.7331	0.7618	0.7322	0.7624	0.8660	0.7921	0.8257	0.7303	0.8048	0.8670	0.7707	0.7280	0.7533
	40%	0.8050	0.8150	0.8289	0.8161	0.8390	0.8453	0.8530	0.9097	0.8146	0.8101	0.8883	1.0000	**0.8017**	0.8207
	50%	**0.8374**	0.8882	0.8734	0.8783	0.8884	0.8743	0.9090	0.8913	0.8796	0.8689	0.9050	1.0000	0.8513	0.8983
Health	0%	0.3770	0.3847	0.3845	0.3834	0.4491	0.5800	0.5552	0.4357	**0.3622**	0.3646	0.8570	0.3913	0.4791	0.3701
	15%	**0.4436**	0.4617	0.4843	0.4437	0.4933	0.6202	0.5899	0.5810	0.4724	0.4943	0.5970	0.5277	0.4963	0.4650
	30%	**0.5531**	0.6010	0.5977	0.5924	0.6273	0.6707	0.6973	0.6830	0.5651	0.6556	0.7383	0.6210	0.5927	0.5860
	40%	0.6783	0.6831	0.6962	0.6861	0.6798	0.7433	0.7249	0.8290	0.7179	0.7087	0.7453	1.0000	**0.6277**	0.6750
	50%	0.7123	0.7407	0.7381	0.7397	0.7464	0.7557	0.7923	0.8277	0.7641	0.7367	0.6890	1.0000	**0.6867**	0.7310
Image	0%	0.7489	0.7379	0.6708	0.7425	0.7058	0.7621	0.7575	0.6863	0.7470	0.7425	0.7125	0.7035	**0.6550**	0.7100
	15%	0.7229	0.7400	0.6625	0.7266	0.7042	0.7517	0.7138	**0.6575**	0.6946	0.7558	0.6825	0.7250	0.6650	0.7138
	30%	0.6468	0.6958	0.6471	0.6779	0.6704	0.6892	0.7258	0.6800	0.6975	0.7375	0.6738	0.6875	**0.6125**	0.7000
	40%	0.6000	0.6338	0.6112	0.6308	0.6229	0.6488	0.6504	0.6600	0.6208	0.6242	0.6400	1.0000	**0.5925**	0.6550
	50%	0.5225	0.5817	0.5504	0.5704	0.5288	0.5333	0.5813	0.5400	0.5513	0.5558	0.5563	1.0000	**0.5188**	0.5300
langua	0%	0.9113	0.9102	0.9142	**0.9039**	0.9360	0.9588	0.9777	0.9846	0.9176	0.9222	0.9726	0.9636	0.9451	0.9277
	15%	0.9413	0.9451	**0.9353**	0.9451	0.9468	0.9760	0.9691	0.9726	0.9451	0.9823	0.9605	0.9417	0.9468	0.9691
	30%	**0.9519**	0.9617	0.9646	0.9651	0.9674	0.9674	0.9794	0.9760	0.9703	0.9811	0.9811	0.9605	0.9623	0.9691
	40%	0.9668	0.9714	0.9668	0.9634	0.9657	**0.9537**	0.9720	0.9777	0.9680	0.9691	0.9931	1.0000	0.9623	0.9640
	50%	0.9657	0.9708	0.9668	0.9708	0.9668	0.9640	0.9766	0.9760	0.9714	0.9663	0.9760	1.0000	**0.9571**	0.9674
Medica	0%	0.2446	0.2190	**0.2114**	0.2225	0.3657	0.2933	0.5575	0.8363	0.2838	0.2847	0.7826	0.3074	0.3018	0.2529
	15%	**0.3955**	0.4586	0.4305	0.3972	0.5013	0.5533	0.7076	0.8389	0.4876	0.4773	0.9872	0.4962	0.4629	0.3969
	30%	0.7327	0.8159	**0.6743**	0.7826	0.8491	0.9156	0.9011	0.9105	0.7724	0.8406	0.9974	0.7980	0.7621	0.7146
	40%	**0.7971**	0.8269	0.8542	0.8397	0.8517	0.9156	0.8559	0.8338	0.8252	0.8312	0.9974	1.0000	0.8184	0.8414
	50%	0.9122	0.9156	0.9028	0.9173	0.9156	0.9156	0.9284	0.9233	0.9233	0.9173	0.9156	1.0000	**0.8977**	0.9079
Recrea	0%	0.5790	0.5798	0.5824	0.5820	0.6497	0.6756	0.7510	0.6630	0.5800	0.5793	0.8270	**0.5760**	0.6969	0.6023
	15%	**0.6456**	0.6566	0.6959	0.6633	0.7002	0.8096	0.8151	0.8077	0.7011	0.6978	0.8037	0.6920	0.7227	0.7117
	30%	**0.7816**	0.8138	0.8086	0.8044	0.8181	0.8710	0.8626	0.8440	0.8144	0.8236	0.8550	0.7833	0.8093	0.8000
	40%	**0.8333**	0.8481	0.8441	0.8484	0.8513	0.9079	0.8727	0.8723	0.8401	0.8513	0.8647	1.0000	0.8413	0.8483
	50%	0.8470	0.8542	0.8500	0.8589	0.8591	0.8577	0.8753	0.8687	0.8581	0.8564	0.8830	1.0000	**0.8340**	0.8657
Refere	0%	0.5298	**0.5251**	0.5414	0.5368	0.5702	0.6797	0.6431	0.6243	0.5381	0.5420	0.9520	0.5537	0.6254	0.5650
	15%	0.5914	0.5717	0.5754	**0.5546**	0.6104	0.7957	0.6911	0.7263	0.6196	0.6280	0.7533	0.5887	0.6193	0.6073
	30%	**0.6580**	0.7064	0.6884	0.6697	0.7147	0.7232	0.7883	0.6840	0.6972	0.7541	0.7977	0.6820	0.6763	0.6893
	40%	0.7630	0.7247	0.7433	0.7521	0.7744	0.8410	0.8464	0.7240	0.7427	0.7273	0.8710	1.0000	**0.6967**	0.8213
	50%	0.8623	0.9178	0.8972	0.8997	0.8890	0.9077	0.9199	0.9330	0.8830	0.9029	0.8893	1.0000	**0.8107**	0.8377
Scene	0%	0.3411	0.3312	0.3059	0.3423	0.3001	0.3645	0.3060	0.4418	0.3610	0.3534	0.4491	0.3528	**0.2775**	0.3112
	15%	0.3821	0.3659	0.3981	0.3950	0.3583	0.3902	**0.3129**	0.4563	0.4695	0.4056	0.4584	0.3867	0.3378	0.5270
	30%	**0.5031**	0.5336	0.5662	0.5360	0.5211	0.5665	0.5430	0.5624	0.5974	0.6216	0.5884	0.5509	0.5031	0.6528
	40%	0.6532	0.7166	0.7148	0.7225	0.6396	0.7339	0.6428	0.6674	0.6677	0.6701	0.6486	1.0000	**0.6091**	0.6996
	50%	0.7734	0.7703	0.7814	0.7789	0.7762	0.7963	0.8004	0.7817	0.7886	0.7921	0.8004	1.0000	**0.7661**	0.7827
Scien	0%	**0.6856**	0.7017	0.7271	0.7190	0.7886	0.9177	0.8590	0.8487	0.7244	0.7106	0.9663	0.7157	0.8248	0.7277
	15%	**0.7560**	0.7825	0.8291	0.7834	0.8044	0.9140	0.8946	0.9040	0.8104	0.8188	0.9097	0.8243	0.8183	0.8363
	30%	**0.8475**	0.8717	0.8752	0.8717	0.8771	0.9427	0.8999	0.9127	0.8861	0.9059	0.9483	0.8917	0.8623	0.8620
	40%	**0.8406**	0.9073	0.9116	0.9063	0.9099	0.8567	0.9151	0.9363	0.9119	0.9032	0.9140	1.0000	0.8767	0.9267
	50%	0.9160	0.9283	0.9143	0.9280	0.9181	0.8983	0.9429	0.9530	0.9328	0.9339	0.9257	1.0000	**0.8910**	0.9207
Slashd	0%	0.5725	**0.5487**	0.5738	0.5509	0.7691	0.7873	0.8488	0.8955	0.5522	0.5893	0.7255	0.6011	0.7136	0.5965
	15%	**0.6139**	0.7444	0.7778	0.7048	0.7830	0.8340	0.8724	0.9180	0.7535	0.7843	0.8783	0.8981	0.7725	0.7976
	30%	0.8369	0.9138	0.9204	0.9007	0.9206	0.9841	0.9614	0.9550	0.9048	0.9583	0.9901	0.9306	0.8452	**0.8089**
	40%	0.9129	0.9680	0.9237	0.9577	0.9405	0.9974	0.9751	0.9742	0.9557	0.9641	0.9815	1.0000	**0.9107**	0.9597
	50%	0.9720	0.9819	**0.9680**	0.9850	0.9850	0.9854	0.9868	0.9888	0.9848	0.9863	0.9848	1.0000	0.9795	0.9841
social	0%	**0.3710**	0.4236	0.4108	0.4228	0.4937	0.8333	0.6029	0.4397	0.4102	0.3951	0.9730	0.3970	0.4977	0.4023
	15%	**0.5109**	0.5146	0.5259	0.5292	0.5664	0.8583	0.6812	0.7747	0.5236	0.5410	0.6657	0.5540	0.5613	0.7887
	30%	0.7938	0.8234	0.8052	0.8123	0.8433	0.9047	0.8892	0.8750	0.8259	0.9072	0.8273	0.8703	0.7910	**0.7837**
	40%	0.9297	0.9374	0.9284	0.9352	0.9073	0.9728	0.9533	0.9597	0.9379	0.9408	0.9670	1.0000	**0.8557**	0.9457
	50%	0.9467	0.9613	0.9540	0.9531	0.9549	0.9583	0.9618	0.9690	0.9583	0.9584	0.9620	1.0000	**0.9440**	0.9510
societ	0%	**0.5217**	0.5438	0.5477	0.5384	0.5869	0.6442	0.7049	0.5980	0.5448	0.5380	0.9437	0.5233	0.6450	0.5583
	15%	0.6320	0.6305	0.6504	0.6405	0.6624	0.6673	0.7423	0.6720	0.6510	0.6781	0.6993	0.6473	**0.6077**	0.6497
	30%	0.8723	0.8894	0.8659	0.8784	0.8909	0.9297	0.8977	0.9270	0.8870	0.9001	0.9020	0.8810	**0.8283**	0.8767
	40%	0.9167	0.9182	0.9183	0.9173	**0.8770**	0.9133	0.9350	0.9377	0.9162	0.9153	0.9130	1.0000	0.8780	0.9170
	50%	0.9137	0.9243	0.9242	0.9228	0.9249	0.9297	0.9336	0.9417	0.9240	0.9241	**0.9127**	1.0000	0.9180	0.9220
Yeast	0%	0.4354	0.4053	0.3954	0.3954	0.3810	0.3723	0.6404	0.4493	0.3954	0.3954	0.3654	0.4590	**0.3427**	0.4121
	15%	0.4823	0.4692	0.4545	0.4585	0.4997	0.5166	0.4921	0.6201	0.5041	0.4696	0.6449	0.4516	**0.3882**	0.5714
	30%	0.4715	0.5307	0.5062	0.5555	0.5694	0.5252	0.6756	0.5580	0.5801	0.5828	0.6025	0.5590	**0.3551**	0.5849
	40%	0.5642	**0.5183**	0.5631	0.5594	0.5973	0.5925	0.6556	0.6511	0.5880	0.5925	0.6801	1.0000	0.5404	0.5735
	50%	0.5645	0.5714	0.5745	0.5794	0.5852	0.5977	0.6335	0.6356	0.6056	**0.5625**	0.6998	1.0000	0.5714	0.6791

Table 8
Performance comparison of different feature selection methods on the NUS-WIDE dataset.

Method	MIC	MAC	HML	ZOL	ONE
RLBMLFS	**0.3475**	**0.0831**	**0.0609**	0.9920	**0.5551**
RFSFS	0.3223	0.0764	0.0615	0.9924	0.5761
LMLFS	0.3370	0.0802	0.0613	0.9923	0.5643
SRFS	0.3171	0.0777	0.0621	0.9930	0.5824
SPLDG	0.3281	0.0756	0.0624	0.9932	0.5792
DRMFS	0.3394	0.0783	0.0615	**0.9908**	0.5676
MLFS-GL	0.2923	0.0629	0.0635	0.9944	0.6120
MRDM	0.3153	0.0548	0.0660	0.9954	0.6096
SSFS	0.3327	0.0741	0.0618	0.9930	0.5798
SCMFS	0.3325	0.0659	0.0642	0.9935	0.5854
SCLS	0.2853	0.0557	0.0652	0.9957	0.6411
SCNMF	0.3393	0.0778	0.0613	0.9918	1.0000
SLCLR	0.3416	0.0797	0.0618	0.9926	0.5735
GNN-DE	0.3088	0.0774	0.0620	0.9940	0.6081

Table 9
Friedman test results for performance comparison under 30% label noise.

Evaluation metric	F_F	Critical value($\boldsymbol{\alpha = 0.05}$)
Micro-F1	27.43	
Macro-F1	30.37	
Hamming Loss	15.00	1.80
Zero-One Loss	22.72	
One-Error	27.42	

sponding column) while keeping the same features largely inactive for other labels, consistent with selecting a small set of label-relevant cues. By comparison, both baselines assign noticeable weights to a broader set of features that are also activated across other labels, reflecting less selective and less semantically grounded feature attribution. Taken together, the clearer block structure, reduced cross-label sharing, and more concentrated label-specific activations indicate that RLBMLFS selects more meaningful features than RFSFS and SCNMF.

4.8. Comparing learned and static label correlation

To evaluate the effect of learning on label-dependency structure, Fig. 16 compares the cosine-similarity prior $\boldsymbol{C}_0$ with the correlation matrix $\boldsymbol{C}$ learned by our model for the Arts and Recreation datasets. A clear qualitative pattern emerges: $\boldsymbol{C}_0$ is relatively diffuse (dense), with many small-to-moderate off-diagonal activations spread across the matrix, consistent with co-occurrence statistics that assign nontrivial similarity to a wide range of label pairs, even when those relationships are weak or incidental. In contrast, the learned matrix $\boldsymbol{C}$ is noticeably sharper and more selective, concentrating its mass into a smaller set of stronger interactions while pushing most other entries toward near-zero. This shift is important because multi-label prediction typically benefits from targeted dependency modeling: dense, low-level couplings can act like “background leakage,” encouraging error propagation (a mistake on one label can weakly bias many others) and reducing interpretability, whereas a sparse, high-contrast structure promotes structured information flow only among the most informative label neighborhoods. Put differently, the model uses supervision to transform a generic similarity heuristic into a task-aligned dependency graph, keeping only relationships that consistently help reduce the loss, which both improves robustness (especially under label imbalance) and yields a more interpretable set of label interactions. The fact that this refinement appears consistently across Arts and Recreation suggests the behavior is not an artifact of a single domain, but a stable property of the proposed learning mechanism: it suppresses spurious correlations from raw co-occurrence and emphasizes the few dependencies that meaningfully support joint multi-label inference.

To highlight the advantages of the learned label correlation, we compare the learned label correlation matrix with the static label correlation matrix. The learned correlation dynamically captures nuanced relationships between labels and mitigates the influence of noisy labels, leading to more accurate predictions and improved robustness. In contrast, the static correlation relies on predefined, fixed relationships that fail to reflect the complexities of the data, resulting in suboptimal performance. Here, we present a comparative analysis, as illustrated in Fig. 17, conducted on the Arts, Emotions, Entertainment, Health, Science, Scene, Yeast, Recreation, and Slashdot datasets using four evaluation metrics: Micro-F1, Macro-F1, HML, and ZOL.

In these Figures, the phrase "RLBMLFS" refers to the method that employs dynamic label correlation, while "ST-RLBMLFS" denotes the method that uses static label correlation. The Figures depicting the learned label correlations reveal a more refined and adaptive understanding of label relationships. Unlike static label correlations, the learned ones adjust to intrinsic data patterns, uncovering deeper and more meaningful dependencies between labels. This adaptability enables the model to move beyond fixed, predefined relationships and discover task-specific structures. Consequently, the learned label correlations yield a more nuanced and context-aware representation of inter-label dependencies, thereby enhancing the model’s capacity to identify relevant features and handle complex label interactions.

4.9. Convergence

In this subsection, we evaluate the convergence behavior of the proposed RLBMLFS model (28) using the Arts, Entertainment, and Society datasets. Algorithm 1 was executed for 70 iterations on each dataset, and the objective function value was plotted against the number of iterations in Fig. 18 to illustrate the convergence process. The results indicate that the objective function value decreases rapidly and steadily during the initial iterations, showing that the method quickly approaches an optimal solution. In the later iterations, the value changes only marginally, confirming that the method has reached a near-optimal solution. These observations demonstrate that the proposed model achieves efficient and reliable convergence.

4.10. Running time

To evaluate the computational efficiency of the compared methods, Table 12 summarizes the running times of each method. As shown, RLBMLFS requires relatively higher running time due to its richer modeling, which includes robust log-based loss, dynamic label correlation, relevance-redundancy regularization, and structured sparsity constraints. RFSFS and SCNMF incorporate label correlation, sparsity, and redundancy or similarity preservation, resulting in moderate running time. SLCLR employs sparse constraints, Laplacian rank regularization, and dynamic graph-based manifold learning, while GNN-DE relies on graph neural network computations, leading to higher computational cost on larger datasets. Despite this, RLBMLFS consistently achieves superior performance and stronger robustness to label noise, demonstrating a favorable trade-off between computational cost and effectiveness.

Table 10
Ablation study of RLBMLFS with 30% label noise, comparing Frobenius norm and Hx loss, and assessing each regularizer via One-Error and Hamming Loss. Best results are **bold**, second-best underlined.

Dataset	One-Error						Hamming Loss					
	I	II	III	IV	V	VI	I	II	III	IV	V	VI
Arts	0.9004	0.8830	0.8570	0.8591	0.8445	**0.8363**	0.1720	0.1677	0.1528	0.1535	0.1527	**0.1367**
Education	0.8731	0.8480	0.7991	0.8173	0.7881	**0.7881**	0.1483	0.1404	0.1315	0.1379	0.1229	**0.1229**
Health	0.6928	0.6455	0.6426	0.6148	0.6038	**0.5531**	0.1530	0.1480	0.1470	0.1298	0.1258	**0.1236**
Recreation	0.8328	0.8096	0.8014	0.8018	0.7998	**0.7816**	0.1742	0.1662	0.1544	0.1487	0.1470	**0.1396**
Reference	0.7624	0.7250	0.7200	0.7146	0.7145	**0.6580**	0.1308	0.1345	0.1220	0.1172	0.1188	**0.1120**
Scene	0.5676	0.5738	0.5483	0.5327	0.5509	**0.5031**	0.2423	0.2396	0.2240	0.2141	0.2216	**0.1953**
Science	0.8998	0.9005	0.8786	0.8800	0.8582	**0.8475**	0.1530	0.1150	0.1202	0.1364	0.1224	**0.1171**
Slashdot	0.9937	0.9677	0.9212	0.8988	0.8964	**0.8369**	0.1763	0.1591	0.1551	0.1466	0.1455	**0.1179**
Social	0.8754	0.8726	0.8570	0.8358	0.8288	**0.7938**	0.1409	0.1306	0.1296	0.1150	0.1084	**0.1020**
Society	0.9016	0.8942	0.8920	0.8901	0.8861	**0.8723**	0.1659	0.1550	0.1540	0.1528	0.1453	**0.1327**
Yeast	0.6101	0.5910	0.5797	0.6050	0.5439	**0.4715**	0.3248	0.3121	0.3111	0.3029	0.3154	**0.2685**

Table 11
Ablation study of RLBMLFS with 30% label noise, comparing Frobenius norm and Hx loss, and assessing each regularizer via Micro-F1 and Macro-F1. Best results are **bold**, second-best underlined.

Dataset	Micro-F1						Macro-F1					
	I	II	III	IV	V	VI	I	II	III	IV	V	VI
Arts	0.1453	0.1665	0.1727	0.1736	0.1789	**0.1881**	0.0941	0.0968	0.1041	0.1019	0.1098	**0.1124**
Education	0.1031	0.1323	0.1587	0.1399	0.1701	**0.1736**	0.0509	0.0599	0.0705	0.0667	0.0814	**0.0814**
Health	0.1857	0.2239	0.2308	0.2492	0.2470	**0.2992**	0.0730	0.0920	0.0920	0.1028	0.1027	**0.1135**
Recreation	0.1660	0.1736	0.1735	0.1762	0.1875	**0.1973**	0.1093	0.1113	0.1176	0.1212	0.1276	**0.1318**
Reference	0.1550	0.1601	0.1628	0.1761	0.1807	**0.2134**	0.0489	0.0624	0.0635	0.0631	0.0671	**0.0753**
Scene	0.4267	0.4363	0.4513	0.4630	0.4496	**0.4923**	0.4000	0.3965	0.4171	0.4164	0.4145	**0.4536**
Science	0.0978	0.1004	0.1083	0.1095	0.1187	**0.1384**	0.0419	0.0590	0.0587	0.0628	0.0712	**0.0770**
Slashdot	0.0982	0.1233	0.1275	0.1869	0.1738	**0.2027**	0.0732	0.0774	0.0917	0.1222	0.1109	**0.1271**
Social	0.1390	0.1715	0.1741	0.1883	0.1734	**0.2399**	0.0488	0.0860	0.0862	0.0878	0.0885	**0.0891**
Society	0.1965	0.1934	0.1986	0.1968	0.1996	**0.2229**	0.0891	0.0842	0.0913	0.0907	0.0923	**0.1028**
Yeast	0.3908	0.4363	0.4513	0.4630	0.4496	**0.5555**	0.1917	0.2625	0.3034	0.2914	0.3050	**0.3356**

Table 12
Running time (seconds) of different methods.

Datasets	RLBMLFS	RFSFS	SCNMF	SLCLR	GNN-DE	SRFS
Emotions	0.1565	0.0720	0.0270	0.2346	3.0630	0.0819
Scene	1.7993	0.4267	0.1649	2.3031	13.6656	0.4077
Language Log	18.5150	3.8665	2.1827	20.536	80.2416	4.6947

5. Conclusion

In this paper, we proposed a Robust Log-Based Multi-Label Feature Selection (RLBMLFS) method for effective dimensionality reduction in noisy multi-label data. To overcome the limitations of existing $L_{2,1}$-norm-based approaches, namely their sensitivity to large noise and dominance by redundant or large-magnitude features, we introduced an Hx robust loss function and a dynamic label correlation mechanism to enhance robustness in feature-label modeling. Furthermore, a logarithmic penalty with $L_{2,\log}$ pseudo-norm regularization was employed to promote sparsity while mitigating the influence of dominant feature weights. An efficient alternating optimization algorithm was developed to solve the resulting objective. Extensive experiments on multiple benchmark datasets demonstrate that RLBMLFS consistently outperforms state-of-the-art methods, particularly in noisy multi-label settings. While the RLBMLFS has demonstrated several strengths and advantages, there are still limitations that need to be addressed to further enhance its performance and applicability in future work. The model depends on two parameters to regulate relevance and redundancy, which may require appropriate parameter selection to avoid unnecessary redundancy or the loss of informative labels. Additionally, the current optimization process may exhibit relatively slow convergence in some cases. Incorporating more efficient optimization algorithms and acceleration strategies could potentially improve convergence speed and overall computational efficiency. Moreover, the proposed model could potentially be extended to learn not only label correlations but also feature correlations, enabling it to capture deeper relationships between input features. Finally, we plan to extend the proposed model to address the multi-view multi-label problem, enhancing its applicability to more complex scenarios.

CRediT authorship contribution statement

Mohammad Faraji: Writing – original draft, Visualization, Software, Methodology, Investigation; **Amjad Seyedi:** Writing – review & editing, Methodology, Investigation, Conceptualization; **Fardin Akhlaghian Tab:** Writing - review & editing, Validation, Supervision, Conceptualization.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Amjad Seyedi reports financial support was provided by the European Research Council. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

Amjad Seyedi acknowledges the support by the European Union (ERC consolidator, eLinoR, no 101085607).

References

[1] W. Liu, H. Wang, X. Shen, I.W. Tsang, The emerging trends of multi-label learning, IEEE Trans. Pattern Anal. Mach. Intell. 44 (11) (2022) 7955–7974.
[2] Z. Sun, H. Xie, J. Liu, Y. Yu, Multi-label feature selection via adaptive dual-graph optimization, Expert Syst. Appl. 243 (2024) 122884.
[3] L. Hu, Y. Li, W. Gao, P. Zhang, J. Hu, Multi-label feature selection with shared common mode, Pattern Recognit. 104 (2020) 107344.
[4] H. Dong, W. Wang, K. Huang, F. Coenen, Automated social text annotation with joint multilabel attention networks, IEEE Trans. Neural Netw. Learn. Syst. 32 (5) (2020) 2224–2238.
[5] X. Cai, F. Nie, H. Huang, Exact top-k feature selection via L2,0-Norm constraint, in: PInternational Joint Conference on Artificial Intelligence, 2013, p. 1240-1246.
[6] D. Rodrigues, L.A.M. Pereira, R.Y.M. Nakamura, K.A.P. Costa, X.-S. Yang, A.N. Souza, J.P. Papa, A wrapper approach for feature selection based on bat algorithm and optimum-Path forest, Expert. Syst. Appl. 41 (5) (2014) 2250–2258.
[7] S.-B. Chen, Y.-M. Zhang, C.H.Q. Ding, J. Zhang, B. Luo, Extended adaptive Lasso for multi-class and multi-label feature selection, Knowl. Based Syst. 173 (2019) 28–36.
[8] C. Tang, M. Bian, X. Liu, M. Li, H. Zhou, P. Wang, H. Yin, Unsupervised feature selection via latent representation learning and manifold regularization, Neural Netw. 117 (2019) 163–178.
[9] J. Hu, Y. Li, W. Gao, P. Zhang, Robust multi-label feature selection with dual-graph regularization, Knowl. Based Syst. 203 (2020) 106126.
[10] Y. Li, J. Hu, W. Gao, Robust multi-label feature selection with shared label enhancement, Knowl. Inf. Syst. 64 (12) (2022) 3343–3372.
[11] N. Gao, S.-J. Huang, S. Chen, Multi-label active learning by model guided distribution matching, Front. Comput. Sci. 10 (5) (2016) 845–855.
[12] S.-J. Huang, S. Chen, Z.-H. Zhou, Multi-label active learning: query type matters, in: International Joint Conference on Artificial Intelligence, 2015, pp. 946–952.
[13] M. Faraji, S.A. Seyedi, F. Akhlaghian Tab, R. Mahmoodi, Multi-label feature selection with global and local label correlation, Expert Syst. Appl. 246 (2024) 123198.
[14] Y. Li, L. Hu, W. Gao, Multi-label feature selection via robust flexible sparse regularization, Pattern Recognit. 134 (2023) 109074.
[15] Y. Li, L. Hu, W. Gao, Robust sparse and low-redundancy multi-label feature selection with dynamic local and global structure preservation, Pattern Recognit. 134 (2023) 109120.
[16] Q. Wang, X. He, X. Jiang, X. Li, Robust bi-stochastic graph regularized matrix factorization for data clustering, IEEE Trans. Pattern Anal. Mach. Intell. 44 (1) (2022) 390–403.
[17] M. Radovic, M. Ghalwash, N. Filipovic, Z. Obradovic, Minimum redundancy maximum relevance feature selection approach for temporal gene expression data, BMC Bioinformatics 18 (1) (2017) 9.
[18] L. Yu, H. Liu, Efficient feature selection via analysis of relevance and redundancy, J. Mach. Learn. Res. 5 (Oct) (2004) 1205–1224.
[19] Z. Zhao, L. Wang, H. Liu, Efficient spectral feature selection with minimum redundancy, in: Proceedings of the AAAI Conference on Artificial Intelligence, 24, 2010, pp. 673–678.
[20] G.H. John, R. Kohavi, K. Pfleger, Irrelevant features and the subset selection problem, in: Machine Learning Proceedings 1994, Elsevier, 1994, pp. 121–129.
[21] W. Gao, Y. Li, L. Hu, Multilabel feature selection with constrained latent structure shared term, IEEE Trans. Neural Netw. Learn. Syst. 34 (3) (2023) 1253–1262.
[22] R. Huang, Z. Wu, Multi-label feature selection via manifold regularization and dependence maximization, Pattern Recognit. 120 (2021) 108149.
[23] Y. Li, X. Wang, X. Yang, W. Gao, W. Ding, T. Li, Fusion-enhanced multi-label feature selection with sparse supplementation, Inf. Fusion 117 (2025) 102813.
[24] F. Nie, H. Huang, X. Cai, C. Ding, Efficient and robust feature selection via joint l2,1-Norms minimization, in: Advances in Neural Information Processing Systems, 23, 2010, pp. 1813–1821.
[25] L. Jian, J. Li, K. Shu, H. Liu, Multi-label informed feature selection, in: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, 2016, pp. 1627—1633.
[26] C. Peng, Y. Zhang, Y. Chen, Z. Kang, C. Chen, Q. Cheng, Log-based sparse nonnegative matrix factorization for data representation, Knowl. Based Syst. 251 (2022) 109127.
[27] J. Lee, D.-W. Kim, SCLS: multi-label feature selection based on scalable criterion for large label set, Pattern Recognit. 66 (2017) 342–352.
[28] P. Zhang, G. Liu, W. Gao, Distinguishing two types of labels for multi-label feature selection, Pattern Recognit. 95 (2019) 72–82.
[29] P. Zhu, Q. Xu, Q. Hu, C. Zhang, H. Zhao, Multi-label feature selection with missing labels, Pattern Recognit. 74 (2018) 488–502.
[30] Z. He, Y. Lin, Z. Lin, C. Wang, Multi-label feature selection via similarity constraints with non-negative matrix factorization, Knowl. Based Syst. 297 (2024) 111948.
[31] Y. Wu, J. Bai, Sparse low-redundancy multi-label feature selection with constrained laplacian rank, Int. J. Mach. Learn. Cybern. 16 (1) (2025) 549–566.
[32] Y. Li, L. Hu, W. Gao, Multi-label feature selection via robust flexible sparse regularization, Pattern Recognit. 134 (2023) 109074.
[33] Y. Li, L. Hu, W. Gao, Label correlations variation for robust multi-label feature selection, Inf. Sci. 609 (2022) 1075–1097.
[34] N. Pan, Multilabel feature selection using graph neural networks and differential evolution optimization, Sci. Rep. 15 (1) (2025) 44349.
[35] D. Kong, C. Ding, H. Huang, Robust nonnegative matrix factorization using l21-norm, in: Proceedings of the 20th ACM International Conference on Information and Knowledge Management, 2011, pp. 673–682.
[36] M. Mozafari, S.A. Seyedi, R. Pir Mohammadiani, F. Akhlaghian Tab, Unsupervised feature selection using orthogonal encoder-decoder factorization, Inf. Sci. 663 (2024) 120277.
[37] S. Soleymanbaigi, A. Seyedi, F. Akhlaghian Tab, F. Daneshfar, Encoder-Decoder nonnegative matrix factorization with β-divergence for data clustering, Pattern Recognit. 171 (2026) 112211.
[38] W. Barkhoda, A. Seyedi, N. Gillis, F. Akhlaghian Tab, Distributionally robust nonnegative matrix factorization with self-paced adaptive multi-loss fusion, Inf. Sci. 728 (2026) 122823.
[39] W. Barkhoda, A. Seyedi, N. Gillis, F. Akhlaghian Tab, Instance-wise distributionally robust nonnegative matrix factorization, Pattern Recognit. 169 (2026) 111732.
[40] H. Wang, H. Huang, C. Ding, Image annotation using multi-label correlated Green's function, in: 2009 IEEE 12th International Conference on Computer Vision, 2009, pp. 2029–2034.
[41] J. Huang, F. Nie, H. Huang, C. Ding, Robust manifold nonnegative matrix factorization, ACM Transactions on Knowledge Discovery from Data (TKDD) 8 (3) (2014) 1–21.
[42] R. Zhang, F. Nie, X. Li, Self-weighted supervised discriminative feature selection, IEEE Trans. Neural Netw. Learn. Syst. 29 (8) (2018) 3913–3918.
[43] N. Jabari, A. Seyedi, R. Mahmoodi, F.a. Tab, Contrastive calibration on consensus and complementary multi-view representations, Pattern Recognit. 176 (2026) 113291.
[44] S. Moradi, A. Seyedi, W. Barkhoda, F.a. Tab, Robust asymmetric encoder-Decoder nonnegative matrix factorization for hyperspectral anomaly detection, Neurocomputing 683 (2026) 133349.
[45] S. Ghodsi, A. Seyedi, T.L. Quy, F. Karimi, E. Ntoutsi, A deep latent factor graph clustering with fairness-utility trade-off perspective, in: The 13th IEEE International Conference on Big Data (IEEE BigData), 2025.
[46] Y. Zhang, J. Tang, Z. Cao, H. Chen, Sparse multi-label feature selection via pseudo-label learning and dynamic graph constraints, Inf. Fusion 118 (2025) 102975.
[47] G. Doquire, M. Verleysen, Mutual information-based feature selection for multilabel classification, Neurocomputing 122 (2013) 148–155.
[48] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, Y. Zheng, NUS-WIDE: A real-world web image database from national university of Singapore, in: Proceedings of the ACM International Conference on Image and Video Retrieval, Association for Computing Machinery, New York, NY, USA, 2009.
[49] M.-L. Zhang, Z.-H. Zhou, ML-KNN: a lazy learning approach to multi-label learning, Pattern Recognit. 40 (7) (2007) 2038–2048.
[50] J. Liu, Y. Lin, S. Wu, C. Wang, Online multi-label group feature selection, Knowl. Based Syst. 143 (2018) 42–57.
[51] J. Zhang, Z. Luo, C. Li, C. Zhou, S. Li, Manifold regularized discriminative feature selection for multi-label learning, Pattern Recognit. 95 (2019) 136–150.
[52] X. Zhao, Q. Li, Z. Xing, X. Yang, X. Dai, Sparse semi-supervised multi-label feature selection based on latent representation, Complex Intell. Syst. 10 (4) (2024) 5139–5151.
[53] Z. Qin, H. Chen, Y. Mi, C. Luo, S.-J. Horng, T. Li, Multi-label feature selection with adaptive graph learning and label information enhancement, Knowl. Based Syst. 285 (2024) 111363.
[54] A. Mohammadi, S.A. Seyedi, F.A. Tab, R.P. Mohammadiani, Diverse joint nonnegative matrix tri-factorization for attributed graph clustering, Appl. Soft Comput. 164 (2024) 112012.
[55] S.A. Seyedi, S.S. Ghodsi, F. Akhlaghian, M. Jalili, P. Moradi, Self-paced multi-label learning with diversity, in: Asian Conference on Machine Learning, PMLR, 2019, pp. 790–805.
[56] R.L. Iman, J.M. Davenport, Approximations of the critical region of the fbietkan statistic, Commun. Stat. Theory Methods 9 (6) (1980) 571–595.