Pattern Recognition Letters
journal homepage: www.elsevier.com

Multilabel Naïve Bayes Classification Considering Label Dependence

Hae-Cheon Kim[a], Jin-Hyeong Park[a], Dae-Won Kim[a], Jaesung Lee[a,**]

[a]*School of Computer Science and Engineering, Chung-Ang University, Seoul, Republic of Korea*

ABSTRACT

Multilabel classification is the task of assigning relevant labels to an instance, and it has received considerable attention in recent years. This task can be performed by extending a single-label classifier, such as the naïve Bayes classifier, to utilize the useful relations among labels for achieving better multilabel classification accuracy. However, the conventional multilabel naïve Bayes classifier treats each label independently and hence neglects the relations among labels, resulting in degenerated accuracy. We propose a new multilabel naïve Bayes classifier that considers the relations or dependence among labels. Experimental results show that the proposed method outperforms conventional multilabel classifiers.

1. Introduction

Multilabel classification is the task of mapping multiple relevant labels to a given instance, and it is a core technique for well-known applications such as text categorization (Elghazel et al., 2016), image annotation (Wu et al., 2015), and music tag classification (Lee et al., 2019). As multilabel classification can be regarded a generalization of the single-label classification problem, numerous multilabel classifiers have been extended from single-label classifiers (Zhang and Zhou, 2014). For example, the naïve Bayes classifier, which is one of the most representative classifiers (Li and Yang, 2018), was extended to the multilabel naïve Bayes classifier (Zhang et al., 2009).

The dependence among labels can be used to improve the accuracy of multilabel classification (Huang et al., 2015; Zhang and Zhou, 2014). For example, in the weather classification problem, the label *raining* is likely to be coupled with the label *cloudy* and unlikely to be coupled with the label *sunny*. However, conventional multilabel naïve Bayes classification neglects the dependence among labels because it treats each label independently. Thus, unobserved label combinations can be assigned, thereby degenerating multilabel classification accuracy.

In this paper, we propose a new multilabel naïve Bayes algorithm that considers the dependence among labels for the classification process, named MLNB-LD. To achieve this, we derive a new posterior probability estimation method for a multilabel problem based on Bayes' theorem with the strong independence assumption. Experimental results indicate that, MLNB-LD outperforms the multilabel naïve Bayes classifier and other conventional multilabel classifiers.

2. Related works

In multilabel classification studies, the methods that utilize label dependence can be broadly divided into three groups according to how many labels are considered concurrently (Zhang and Zhou, 2014). The first group of classifiers treats each label independently by inferencing a mapping function for each label. For example, Zhang and Zhou (2007) proposed a multilabel k-nearest neighbor classifier that identifies k similar instances from a training set and then determines the relevance of each label. Vens et al. (2008) proposed new multilabel decision trees that consider the label hierarchy in a hierarchical multilabel classification (MLDT). Zhang et al. (2009) extended the conventional naïve Bayes classifier to a multilabel naïve Bayes classifier that estimates the posterior probability for each label independently. In addition, Zhang and Wu (2015) proposed a multilabel classifier that selects a subset of relevant features for each label. Lastly, Luo et al. (2017) introduced a multilabel kernel extreme learning machine (ML-kELM) that calculates the likelihood of each label based on the random weighting scheme and radial basis kernel mapping.

In the second group, multilabel classifiers consider the label dependence between label pairs. For example, Huang et al.

**Corresponding author: Tel.: +82-02-820-5468;
e-mail: curseor@cau.ac.kr (Jaesung Lee)

(2015) proposed a classifier that selects important features for each label and then calculates the similarity between selected feature subsets and label pairs. In addition, Huang et al. (2017) devised a multilabel classifier that uses local positive and negative pairwise label correlation. Jing et al. (2017) introduced semisupervised multilabel classification that applies singular value decomposition for label matrix factorization. Similarly, Kumar et al. (2018) proposed a hierarchical embedding-based multilabel classifier that is based on k-means clustering and low-rank matrix factorization. Zhu et al. (2018) developed multilabel learning with a global and local label correlation (GLOCAL) strategy that used the correlation among labels in the global and local viewpoints using low-rank matrix factorization.

In the third group, the classification process is designed to consider an arbitrary number of labels concurrently. For example, the random k-labelset algorithm creates k label sets by encoding multiple arbitrarily selected labels into a series of single labels. Then, classifiers are trained for each transformed label (Tsoumakas et al., 2010). After prediction is completed, the transformed single labels are recovered to the original multiple labels. The classifier chain approach selects the number of labels to be considered concurrently and chains the prediction model for each label using the prediction of labels in the early stage of the chain to labels in the later stage (Read et al., 2011). This technique was applied to recurrent neural networks to maximize subset accuracy (Nam et al., 2017). Lastly, the k-nearest neighbor classifier was extended to a multilabel classification problem by utilizing fuzzy rough neighborhood consensus and label correlation estimation with the weighted Hamming distance (Vluymans et al., 2018).

Our brief review shows that the multilabel classifiers in the first group take the simplest approach and conventional single-label classifiers can be directly used by treating each label as multiple individual problems. However, this approach inherently neglects the dependence among labels that can be useful for improving multilabel classification accuracy. The classifiers in the third group experience difficulty in significance estimation because they consider a large number of labels simultaneously based on a limited number of training instances. To circumvent both drawbacks, we design a method based on the strategy of the second group, which considers a maximum of two labels concurrently.

3. Proposed method

First, we describe the notation used for deriving the proposed posterior probability estimation method for multilabel classification. Let $\mathcal{X} \subset \mathbb{R}^m$ be the input space and $\mathcal{L} = \{l_1, \ldots, l_n\}$ be the finite set of possible labels. Vector $\mathbf{x} = (x_1, \ldots, x_m)$ represents m features. Vector $\mathbf{y} = (y_1, \ldots, y_n)$ represents n labels, where $y_i \in \mathbb{B}$ is 1 if the i-th label, l_i, is related to a given instance; otherwise, it is 0. Then, a set of multilabeled instances, $(\mathbf{x}, \mathbf{y})$, compose dataset $\mathcal{D}$. In addition, we denote $\mathcal{Y}$ ($|\mathcal{Y}| \leq |\mathcal{D}|$) as a set of the label vectors that are observed from the dataset.

3.1. Derivation

The goal of the multilabel naïve Bayes classifier based on the maximum a posteriori decision rule is to find a hypothesis, $h : \mathbf{x} \rightarrow \mathbf{y}$, where $h(\mathbf{x})$ can be defined as follows:

$$h(\mathbf{x}) = \underset{\mathbf{y} \in \mathcal{Y}}{\arg\max}\, p(\mathbf{y}|\mathbf{x}) = \underset{\mathbf{y} \in \mathcal{Y}}{\arg\max} \frac{p(\mathbf{x}, \mathbf{y})}{p(\mathbf{x})} \tag{1}$$

where $p(\mathbf{y}|\mathbf{x})$ is the conditional probability of $\mathbf{y}$ given $\mathbf{x}$. It is unnecessary to identify the exact value of $p(\mathbf{x})$ because it is the same for all values of $\mathbf{y} \in \mathcal{Y}$. Thus, Eq. (1) can be simplified as follows:

$$\begin{aligned} h(\mathbf{x}) &= \underset{\mathbf{y} \in \mathcal{Y}}{\arg\max} \frac{p(\mathbf{x}, \mathbf{y})}{p(\mathbf{x})} \\ &\propto \underset{\mathbf{y} \in \mathcal{Y}}{\arg\max}\, p(\mathbf{x}, \mathbf{y}) \\ &= \underset{(y_1, \ldots, y_n) \in \mathcal{Y}}{\arg\max} \underbrace{p(x_1, \ldots, x_m, y_1, \ldots, y_n)}_{\text{Part 1}} \end{aligned} \tag{2}$$

The direct calculation of Part 1 of Eq. (2) is unreliable owing to its high dimensionality and a limited number of training instances. Using the chain rule of conditional probability, Part 1 can be rewritten as

$$\begin{aligned} & p(x_1, \ldots, x_m, y_1, \ldots, y_n) \\ =& p(x_1|x_2, \ldots, x_m, y_1, \ldots, y_n) p(x_2, \ldots, x_m, y_1, \ldots, y_n) \\ =& p(x_1|x_2, \ldots, x_m, y_1, \ldots, y_n) p(x_2|x_3, \ldots, x_m, y_1, \ldots, y_n) \\ & \quad p(x_3, \ldots, x_m, y_1, \ldots, y_n) \\ =& \ldots \\ =& p(x_1|x_2, \ldots, x_m, y_1, \ldots, y_n) \cdots p(x_m|y_1, \ldots, y_n) \\ & \quad p(y_1|y_2, \ldots, y_n) \cdots p(y_{n-1}|y_n) p(y_n) \end{aligned} \tag{3}$$

Based on the naïve conditional independence assumption that all features and labels are mutually independent, conditional on label y_n, we have $p(x_i|x_{i+1}, \ldots, x_m, y_1, \ldots, y_n) \approx p(x_i|y_n)$. Under this assumption, Eq. (3) can be expressed as

$$\begin{aligned} & p(x_1, \ldots, x_m, y_1, \ldots, y_n) \\ \approx& p(y_n) p(x_1|y_n) \cdots p(x_m|y_n) \cdots p(y_1|y_n) \cdots p(y_{n-1}|y_n) \\ \approx& p(y_n) \prod_{i=1}^{m} p(x_i|y_n) \prod_{j=1}^{n-1} p(y_j|y_n) \end{aligned} \tag{4}$$

Eq. (4) can be further simplified as follows:

$$\begin{aligned} & p(x_1, \ldots, x_m, y_1, \ldots, y_n) \\ \approx& p(y_n) \prod_{i=1}^{m} p(x_i|y_n) \prod_{j=1}^{n-1} p(y_j|y_n) \\ =& p(y_n) \prod_{i=1}^{m} p(x_i|y_n) \prod_{j=1}^{n} p(y_j|y_n) \end{aligned} \tag{5}$$

Algorithm 1: MLNB-LD($\mathcal{D}, \mathbf{x}$)

Input : $\mathcal{D}, \mathbf{X}$ ▹ Training dataset $\mathcal{D}$, Unseen instances $\mathbf{X}$
Output: $\mathbf{Y}^*$ ▹ Predicted label vectors for $\mathbf{X}$

1 **forall the** $\mathbf{y} \in \mathcal{Y}$ **do**
2 **for** $i \leftarrow 1$ to n **do**
3   $p_{y_i} \leftarrow p(y_i)$;
4   **for** $k \leftarrow 1$ to n **do**
5    $p_{y_k|y_i} \leftarrow p(y_k, y_i)/p_{y_i}$;
6   **end**
7   **forall the** $\mathbf{x} \in \mathbf{X}$ **do**
8   **for** $j \leftarrow 1$ to m **do**
9     $p_{x_j|y_i} \leftarrow p(x_j, y_i)/p_{y_i}$;
10   **end**
11   **end**
12 **end**
13 $\mathcal{S}(\mathbf{y}) \leftarrow \prod_{i=1}^{n} p_{y_i} \prod_{j=1}^{m} p_{x_j|y_i} \prod_{k=1}^{n} p_{y_k|y_i}$ for all $\mathbf{x} \in \mathbf{X}$;
14 **end**
15 $\mathbf{Y}^* \leftarrow \arg\max_{\mathbf{y} \in \mathcal{Y}} \mathcal{S}(\mathbf{y})$ for all $\mathbf{x} \in \mathbf{X}$;

Table 1: Example dataset

Outlook x_1	Temper. x_2	Humidity x_3	Walk y_1	Swim y_2	Tenis y_3
Sunny	Hot	Low	1	0	1
Rainy	Hot	Low	1	1	0
Sunny	Cool	Low	0	1	1
Rainy	Cool	High	0	0	1
Sunny	Cool	High	1	1	0
Rainy	Cool	Low	0	1	0

Table 2: $p(\mathbf{x}|y)$ for $\mathbf{x} = (\text{Sunny}, \text{Hot}, \text{Low})$

$p(\mathbf{x}\|\mathbf{y})$	y_1		y_2		y_3	
	0	1	0	1	0	1
x_1 = Sunny	2/3	1/3	1/2	2/4	1/3	2/3
x_2 = Hot	1/3	2/3	1/2	1/4	1/3	2/3
x_3 = High	1/3	2/3	1/2	2/4	1/3	1/3

Table 3: Probability values of label–label pairs $p(y_k|y_l)$

$y_l \rightarrow$ / $y_k \rightarrow$	y_1		y_2		y_3	
	0	1	0	1	0	1
$y_1 = 0$	1	0	1/2	2/4	1/3	2/3
$y_1 = 1$	0	1	1/2	2/4	2/3	1/3
$y_2 = 0$	1/3	1/3	1	0	0	2/3
$y_2 = 1$	2/3	2/3	0	1	1	1/3
$y_3 = 0$	1/3	2/3	0	3/4	1	0
$y_3 = 1$	2/3	1/3	1	1/4	0	1

which is an estimation of $p(\mathbf{x}, \mathbf{y})$ when focusing on y_n. In addition, Eq. (5) indicates that n estimations can be obtained by considering labels y_1 through y_n; this is written as

$$\begin{aligned} & p(x_1, \ldots, x_m, y_1, \ldots, y_n) \\ \approx & p(y_1) \prod_{i=1}^{m} p(x_i|y_1) \prod_{j=1}^{n} p(y_j|y_1) \\ \approx & \qquad \vdots \\ \approx & p(y_n) \prod_{i=1}^{m} p(x_i|y_n) \prod_{j=1}^{n} p(y_j|y_n) \end{aligned} \tag{6}$$

To determine the value of $p(\mathbf{x}, \mathbf{y})$, we used the geometric mean for aggregating n estimations. As a result, Eq. (6) can be aggregated as follows:

$$\begin{aligned} & p(x_1, \ldots, x_m, y_1, \ldots, y_n) \\ \approx & \Bigg(\prod_{i=1}^{n} p(y_i) \prod_{j=1}^{m} p(x_j|y_i) \underbrace{\prod_{k=1}^{n} p(y_k|y_i)}_{\text{Part 2}} \Bigg)^{\frac{1}{n}} \end{aligned} \tag{7}$$

Part 2 of Eq. (7) indicates that the proposed estimation considers the conditional probability of all label pairs. By replacing Part 1 of Eq. (2) with Eq. (7), we have the following:

$$\begin{aligned} h(\mathbf{x}) &= \arg\max_{\mathbf{y} \in \mathcal{Y}} \left(\prod_{i=1}^{n} p(y_i) \prod_{j=1}^{m} p(x_j|y_i) \prod_{k=1}^{n} p(y_k|y_i) \right)^{\frac{1}{n}} \\ &= \arg\max_{\mathbf{y} \in \mathcal{Y}} \underbrace{\prod_{i=1}^{n} p(y_i) \prod_{j=1}^{m} p(x_j|y_i)}_{\text{Part 3}} \prod_{k=1}^{n} p(y_k|y_i) \end{aligned} \tag{8}$$

In conventional naïve Bayes classification, Part 3 of Eq. (8) is considered to determine the relevance of a given instance to each label; the relevance score is penalized by multiplying $p(y_i)$ and $p(x_j|y_i)$ terms. Eq. (8) shows that MLNB-LD further penalizes the relevance score by multiplying the joint probability value of label pairs conditioned by a label, i.e., $p(y_k|y_i)$ terms, indicating that the score value will decrease considerably when a rare label pair is considered.

Algorithm 1 shows the procedure of the proposed MLNB-LD, which classifies the label set of a given instance set $\mathbf{X}$. The algorithm computes the marginal probability of each label (Line 3) for each label vector, $\mathbf{y} \in \mathcal{Y}$ (Line 1). The algorithm then computes the conditional probabilities of y_k (Line 5) given y_i using the already calculated p_{y_i}. Next, the algorithm computes the conditional probabilities of x_j (Line 9) given y_i for all instances $\mathbf{x} \in \mathbf{X}$. Finally, $\mathcal{S}(\mathbf{y})$, which estimates the posterior probability of $\mathbf{y}$ given $\mathbf{x}$, is calculated and stored (Line 13). These procedures are repeated until all values of $\mathcal{S}(\cdot)$ are computed. Finally, based on the maximum a posteriori rule, the label vector, $\mathbf{y}$, which leads to the maximum value among $\mathcal{S}(\cdot)$, is selected as the predicted label, $\mathbf{y}^*$ (Line 15).

We analyze the time complexity of MLNB-LD based on Algorithm 1. As most processes involve probability estimation, we assume the probability estimation of a feature or a label as a unit cost. For example, the algorithm must incur one unit cost for calculating $p(y_i)$ and two unit costs for calculating $p(y_i, y_j)$. In Line 3, the algorithm incurs one unit cost for calculating $p(y_i)$ and then computes n joint probability values, $2n$ unit cost is incurred to calculate the joint probability between y_i and all label pairs. Then, for all $\mathbf{x} \in \mathbf{X}$, $p(x_j, y_i)$, by incurring a $2m$ unit cost, we neglect the cost of computing $\mathcal{S}(\cdot)$ because it does not involve a probability estimation, indicating that the $1+2n+2m\cdot|\mathbf{X}|$ unit cost is incurred for computing the posterior probability of a label set, $\mathbf{y} \in \mathcal{Y}$. Thus, the algorithm incurs a $(1 + 2n + 2m \cdot |\mathbf{X}|) \cdot |\mathcal{Y}|$ computational cost for instance, set $\mathbf{X}$.

3.2. Toy example

We used the example dataset shown in Table 1 to understand the underlying mechanism of MLNB-LD. This dataset is composed of six instances, three features (Outlook, Temperature, and Humidity), and three labels (Walk, Swim, and Tennis, which are implementable exercises). Specifically, three labels are encoded to the binary label vector (y_1, y_2, y_3). Suppose that we have an unseen instance, $\mathbf{x} = (\text{Sunny}, \text{Hot}, \text{High})$. Here, MLNB-LD must compute a series of probability values to identify the most probable label set. For example, based on the example data, $p(y_1 = 0) = 1/2$, $p(y_1 = 1) = 1/2$, $p(y_2 = 0) = 2/3$, $p(y_2 = 1) = 1/3$, $p(y_3 = 0) = 1/2$, and $p(y_3 = 1) = 1/2$. Tables 2 and 3 show the joint probability values between the feature–label and label–label pairs. In this example, $\mathcal{Y}$ contains five label vector elements, $\mathcal{Y} = \{(0,0,1),(0,1,0),(0,1,1),(1,0,1),(1,1,0)\}$.

Based on Algorithm 1, MLNB-LD computes $\mathcal{S}(\cdot)$, where label $(y_1, y_2, y_3) = (0, 0, 1)$ given $\mathbf{x}$ as follows:

$$
\begin{aligned}
&p(y_1 = 0, y_2 = 0, y_3 = 1 | x_1 = \text{Sunny}, x_2 = \text{Hot}, x_3 = \text{High}) \\
\propto &p(x_1 = \text{Sunny}, x_2 = \text{Hot}, x_3 = \text{High}, y_1 = 0, y_2 = 0, y_3 = 1)
\end{aligned}
$$

Thus, $\mathcal{S}(0, 0, 1)$ is calculated as follows:

$$
\begin{aligned}
&p(x_1 = \text{Sunny}, x_2 = \text{Hot}, x_3 = \text{High}, y_1 = 0, y_2 = 0, y_3 = 1) \\
\approx &p(x_1 = \text{Sunny}|y_1 = 0)p(x_2 = \text{Hot}|y_1 = 0)p(x_3 = \text{High}|y_1 = 0) \\
&p(y_1 = 0)p(y_1 = 0|y_1 = 0)p(y_2 = 0|y_1 = 0)p(y_3 = 1|y_1 = 0) \\
&p(x_1 = \text{Sunny}|y_2 = 0)p(x_2 = \text{Hot}|y_2 = 0)p(x_3 = \text{High}|y_2 = 0) \\
&p(y_2 = 0)p(y_1 = 0|y_2 = 0)p(y_2 = 0|y_2 = 0)p(y_3 = 1|y_2 = 0) \\
&p(x_1 = \text{Sunny}|y_3 = 1)p(x_2 = \text{Hot}|y_3 = 1)p(x_3 = \text{High}|y_3 = 1) \\
&p(y_3 = 1)p(y_1 = 0|y_3 = 1)p(y_2 = 0|y_3 = 1)p(y_3 = 1|y_3 = 1) \\
= &\underbrace{\frac{1}{2}}_{p(y_1=0)} \cdot \underbrace{\frac{2}{3} \cdot \frac{1}{3} \cdot \frac{1}{3}}_{p(\mathbf{x}|y_1=0)} \cdot \underbrace{1 \cdot \frac{1}{3} \cdot \frac{2}{3}}_{p(\mathbf{y}|y_1=0)} \cdot \underbrace{\frac{1}{3}}_{p(y_2=0)} \cdot \underbrace{\frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2}}_{p(\mathbf{x}|y_2=0)} \cdot \underbrace{\frac{1}{2} \cdot 1 \cdot 1}_{p(\mathbf{y}|y_2=0)} \cdot \\
&\underbrace{\frac{1}{2}}_{p(y_3=1)} \cdot \underbrace{\frac{2}{3} \cdot \frac{2}{3} \cdot \frac{1}{3}}_{p(\mathbf{x}|y_3=1)} \cdot \underbrace{\frac{2}{3} \cdot \frac{2}{3} \cdot 1}_{p(\mathbf{y}|y_3=1)} \approx 5.64 \times 10^{-6}
\end{aligned}
$$

Here, $\mathcal{S}(\cdot)$ for a label set can be calculated as zero if any $p(x_j|y_i) = 0$, which is known as the zero-frequency problem. A smoothing technique, such as add-one smoothing, can be used to solve this problem in the real world (Zhang et al., 2009). Finally, (Walk, Swim, Tennis) = (Yes, Yes, No) is selected as the most probable label set for $\mathbf{x} = (\text{Sunny}, \text{Hot}, \text{High})$ because $\mathcal{S}(0, 1, 0) \approx 7.93 \times 10^{-7}$, $\mathcal{S}(0, 1, 1) \approx 1.41 \times 10^{-6}$, $\mathcal{S}(1, 0, 1) \approx 2.82 \times 10^{-6}$, and $\mathcal{S}(1, 1, 0) \approx 1.52 \times 10^{-4}$.

4. Experimental results

4.1. Experimental settings

To conduct the empirical experiments, we used 14 publicly available multilabel datasets that are frequently used in multilabel classification studies (Zhang and Zhou, 2014). The Art, Education, Entertain, Health, Recreation, Reference, Science, Social, and Society datasets (Ueda and Saito, 2003) were obtained from the Yahoo text data collection after removing unnecessary features (Zhang and Wu, 2015). In addition, the Bibtex (Tsoumakas et al., 2010), Enron, and Slashdot (Zhang and Wu, 2015) datasets were obtained for the text categorization tasks. The Corel5k (Zhang and Wu, 2015) dataset was created from annotated images, each containing multiple object segments. The Emotions (Trohidis et al., 2011) dataset was created for the music emotion recognition task. Table 4 presents the characteristics of each dataset. In the first row, *Instances*, *Features*, and *Labels* denote the number of instances, features, and labels, respectively. *Cardinality* and *Density* indicate the average number of labels assigned to each instance and the average occurrences of each label, respectively. *Distinct* denotes the number of unique label sets.

We used four conventional multilabel classifiers to validate the superiority of MLNB-LD against conventional methods. MLNB is an extension of the conventional naïve Bayes classifier, where each label is learned individually (Zhang et al., 2009). In our experiments, the multinomial model was applied after numerical features were categorized by a supervised discretization method (Cano et al., 2016). MLDT adapts predictive clustering trees to induce a single-tree structure for hierarchical multilabel classification (Vens et al., 2008). In our experiments, we used the MLDT with no binary split and the minimum weighted fraction set to two at the whole leaf nodes. The ML-kELM is a single-layered feedforward neural network with random projection and kernel mapping (Luo et al., 2017). Specifically, radial basis kernel mapping based on a Gaussian distribution is used, where the kernel and cost parameter are set as $\sigma = 2^{-2}$ and $C = [2^0, 2^1, 2^2, 2^3]$, respectively. Finally, the multilabel learning approach named GLOCAL that utilizes the correlation among labels from the global and local viewpoints using low-rank matrix factorization is used (Zhu et al., 2018). In our experiments, the threshold values were set as 0.5 and the matrix factorization and cost parameter were set as $k = [5, 10, 15, 20, 25]$ and $\lambda = 1$, respectively.

We used three evaluation measures to compare the quality of multilabel classification results, i.e., Macro F_1, Micro F_1, and Multilabel accuracy. Suppose that a multilabel classifier can output a predicted label vector, $\hat{\mathbf{y}} = h(\mathbf{x})$, for a test instance, $\mathbf{x} \in \mathcal{T}$, where $\hat{\mathbf{y}} = (\hat{y_1}, \cdots, \hat{y_n})$. Then, statistics can be obtained from a contingency table established based on the ground truth for the i-th label, $y_i \in \mathbb{B}$, and the prediction, $\hat{y_i} \in \mathbb{B}$. For example, the *true positive* for the i-th label can be indicated by

$$\text{TP}_i = y_i \cdot \hat{y_i}$$

Similarly, the *false positive*, *true negative*, and *false negative* for the i-th label can be indicated by $\text{FP}_i = (1 - y_i) \cdot \hat{y_i}$, $\text{TN}_i = (1 - y_i) \cdot (1 - \hat{y_i})$, and $\text{FN}_i = y_i \cdot (1 - \hat{y_i})$, respectively.

Table 4: Standard characteristics of used datasets

Name	Domain	Instances	Features	Labels	Cardinality	Density	Distinct
Arts	Text	7,484	1,157	26	1.654	0.064	599
Education	Text	12,030	1,377	33	1.463	0.044	511
Entertain	Text	12,730	1,600	21	1.414	0.067	337
Health	Text	9,205	1,530	32	1.644	0.051	335
Recreation	Text	12,828	1,516	22	1.429	0.065	530
Reference	Text	8,027	1,984	33	1.174	0.036	275
Science	Text	6,428	1,859	40	1.45	0.036	457
Social	Text	12,111	2,618	39	1.279	0.033	361
Society	Text	14,512	1,590	27	1.67	0.062	1,054
Bibtex	Text	7,395	1,836	159	2.402	0.015	2,856
Corel5k	Image	5,000	499	374	3.522	0.009	3,175
Enron	Text	1,702	1,001	53	3.378	0.064	753
Emotions	Music	593	72	6	1.868	0.311	27
Slashdot	Text	3,782	1,079	22	1.181	0.054	156

Table 5: Comparison results in terms of Macro F_1 measure

Dataset	Proposed	MLNB	MLDT	ML-kELM	GLOCAL
Arts	**0.233±0.011**✓	0.225±0.005	0.216±0.024	0.146±0.01	0.057±0.022
Education	**0.157±0.009**✓	0.144±0.005	0.132±0.028	0.139±0.012	0.059±0.019
Entertain	**0.266±0.015**✓	0.251±0.008	0.266±0.013	0.185±0.007	0.097±0.02
Health	**0.227±0.01**✓	0.199±0.005	0.176±0.037	0.181±0.012	0.143±0.021
Recreation	**0.322±0.012**✓	0.279±0.009	0.283±0.013	0.225±0.007	0.079±0.021
Reference	**0.131±0.006**✓	0.127±0.005	0.122±0.035	0.088±0.004	0.072±0.026
Science	**0.147±0.009**✓	0.13±0.005	0.137±0.022	0.085±0.005	0.082±0.054
Social	**0.153±0.01**✓	0.121±0.004	0.147±0.025	0.094±0.003	0.04±0.004
Society	0.164±0.008	0.159±0.004	**0.187±0.017**✓	0.119±0.005	0.031±0.01
Bibtex	**0.23±0.011**✓	0.184±0.005	0.155±0.01	0.158±0.01	0.071±0.007
Corel5k	**0.213±0.015**✓	0.017±0.007	0.141±0.013	0.033±0.011	0.185±0.056
Enron	**0.255±0.028**✓	0.104±0.031	0.223±0.028	0.109±0.015	0.198±0.011
Emotions	0.642±0.031	**0.666±0.024**✓	0.653±0.037	0.589±0.029	0.641±0.028
Slashdot	**0.302±0.012**✓	0.29±0.008	0.301±0.023	0.143±0.015	0.275±0.025
Avg. Rank.	**1.214**	2.714	2.643	3.929	4.5

In addition, the Macro F_1 value for measuring the quality of multilabel classification on $\mathcal{T}$ can be calculated as

$$\text{Macro } F_1 = \frac{1}{|\mathcal{T}|}\sum_{\mathbf{x}\in\mathcal{T}}\left(\frac{1}{n}\sum_{i=1}^{n}\frac{2\text{TP}_i}{2\text{TP}_i + \text{FN}_i + \text{FP}_i}\right)$$

where Macro F_1 evaluates how accurately the classifier can predict the ground truth on average for each test instance. Next, Micro F_1 can be calculated as

$$\text{Micro } F_1 = \frac{1}{|\mathcal{T}|}\sum_{\mathbf{x}\in\mathcal{T}}\frac{2\sum_{i=1}^{n}\text{TP}_i}{2\sum_{i=1}^{n}\text{TP}_i + \sum_{i=1}^{n}\text{FN}_i + \sum_{i=1}^{n}\text{FP}_i}$$

where Micro F_1 evaluates how accurately the classifier predict the ground truth on average for each label. Multilabel accuracy (Mlacc) can be calculated as

$$\text{Mlacc} = \frac{1}{|\mathcal{T}|}\sum_{\mathbf{x}\in\mathcal{T}}\left(\frac{1}{n}\sum_{i=1}^{n}\frac{\text{TP}_i}{\text{TP}_i + \text{FN}_i + \text{FP}_i}\right)$$

where Mlacc outputs the ratio of *true positive* and the summation of the ground truth and positively-predicted labels.

We used the hold-out cross-validation strategy to simulate the real-world performance of each classifier. In a given dataset, 80% of the instances were randomly selected as the training set $\mathcal{D}$, and the remaining 20% were selected as the test set $\mathcal{T}$. The experiment was repeated 30 times for each classifier and dataset, and the average value of each evaluation measure was reported as the multilabel classification performance for comparison. In addition, we used the widely-used Friedman test to compare the performance of multiple classifiers. Based on the average rank of each classifier, the null hypothesis that all classifiers perform equally well was either rejected or accepted. When the null hypothesis was rejected, we performed the Bonferroni–Dunn test to analyze the relative performance among the classifiers. For the Bonferroni–Dunn test, the performances of MLNB-LD and conventional classifiers were regarded as statistically different in 95% if their average ranks

Table 6: Comparison results in terms of Micro F_1 measure

Dataset	Proposed	MLNB	MLDT	ML-kELM	GLOCAL
Arts	**0.423±0.01**✓	0.353±0.008	0.333±0.011	0.258±0.011	0.156±0.055
Education	**0.421±0.01**✓	0.336±0.008	0.37±0.008	0.316±0.007	0.254±0.08
Entertain	**0.442±0.011**✓	0.377±0.011	0.433±0.008	0.323±0.009	0.261±0.044
Health	**0.576±0.009**✓	0.48±0.008	0.545±0.009	0.456±0.013	0.521±0.066
Recreation	**0.441±0.01**✓	0.371±0.012	0.412±0.008	0.297±0.007	0.143±0.038
Reference	**0.45±0.014**✓	0.303±0.008	0.428±0.011	0.267±0.013	0.423±0.086
Science	**0.304±0.013**✓	0.219±0.006	0.225±0.013	0.159±0.008	0.285±0.111
Social	**0.532±0.008**✓	0.346±0.006	0.519±0.01	0.314±0.009	0.456±0.043
Society	0.301±0.007	0.239±0.003	**0.352±0.008**✓	0.27±0.006	0.23±0.041
Bibtex	**0.315±0.011**✓	0.198±0.006	0.179±0.01	0.237±0.012	0.242±0.013
Corel5k	**0.266±0.008**✓	0.097±0.016	0.147±0.005	0.03±0.01	0.244±0.009
Enron	**0.504±0.016**✓	0.24±0.08	0.469±0.015	0.128±0.028	0.415±0.009
Emotions	0.677±0.029	**0.68±0.025**✓	0.668±0.033	0.608±0.026	0.657±0.028
Slashdot	**0.57±0.016**✓	0.557±0.013	0.47±0.015	0.291±0.015	0.444±0.044
Avg. Rank.	**1.143**	3.357	2.5	4.5	3.429

Table 7: Comparison results in terms of Multilabel accuracy measure

Dataset	Proposed	MLNB	MLDT	ML-kELM	GLOCAL
Arts	**0.405±0.01**✓	0.328±0.007	0.319±0.011	0.222±0.01	0.106±0.04
Education	**0.376±0.01**✓	0.32±0.008	0.323±0.007	0.24±0.006	0.169±0.066
Entertain	**0.397±0.008**✓	0.348±0.008	0.35±0.007	0.303±0.007	0.182±0.029
Health	**0.52±0.009**✓	0.476±0.006	0.518±0.01	0.438±0.011	0.458±0.079
Recreation	0.411±0.01	0.343±0.011	**0.412±0.007**✓	0.261±0.006	0.091±0.026
Reference	**0.446±0.014**✓	0.388±0.02	0.427±0.012	0.388±0.015	0.325±0.103
Science	**0.267±0.012**✓	0.215±0.006	0.221±0.013	0.182±0.009	0.209±0.101
Social	**0.544±0.009**✓	0.516±0.009	0.539±0.011	0.454±0.012	0.364±0.047
Society	0.261±0.007	0.202±0.004	**0.31±0.008**✓	0.265±0.007	0.184±0.037
Bibtex	**0.248±0.008**✓	0.192±0.007	0.23±0.011	0.185±0.008	0.192±0.013
Corel5k	**0.181±0.006**✓	0.08±0.03	0.102±0.004	0.02±0.008	0.144±0.005
Enron	0.356±0.015	0.207±0.094	**0.359±0.015**✓	0.076±0.04	0.292±0.009
Emotions	**0.569±0.031**✓	0.559±0.03	0.456±0.033	0.493±0.027	0.532±0.035
Slashdot	**0.554±0.017**✓	0.445±0.014	0.458±0.016	0.25±0.013	0.382±0.036
Avg. Rank.	**1.214**	3.143	2.143	4.357	4.143

over all datasets were larger than one critical difference (CD). In our experiments, the CD is 1.6125 (Demšar, 2006).

4.2. Experimental results

Tables 5–7 show the experimental results obtained using MLNB-LD and the conventional multilabel classifiers on 14 multilabel datasets. They are represented in terms of the average performance with the corresponding standard deviations. The highest performance is shown in bold face and indicated by a check mark (✓). The term ‘Avg. Rank’ at the bottom of each table indicates the average rank for each multilabel classifier over all datasets. Table 8 shows the Friedman statistics and the corresponding critical values of each evaluation measure for each multilabel classifier. We set the significance level as $\alpha = 0.05$. In Figs. 1–3, the CD diagrams illustrate the relative performance of MLNB-LD and the conventional multilabel classifiers. Herein, the average rank of each multilabel classifier is marked along the upper axis, with the higher ranks placed on the left side. We also present the CD from the perspective of MLNB-LD above the graph. This implies that the multilabel classifiers outside the range are significantly different from each other.

Table 8: Friedman statistics and critical value

Evaluation measure	Friedman statistics	Critical value ($\alpha = 0.05$)
Macro F_1	41.5	
Micro F_1	24.9	14.9
Multilabel Accuracy	30.6	

From the results shown in Tables 5–7, it is evident that MLNB-LD outperforms the conventional multilabel classifiers

Table 9: Comparison results of Proposed and MMSE in terms of three evaluation measures

Dataset	Macro F_1 Proposed	Macro F_1 MMSE	Micro F_1 Proposed	Micro F_1 MMSE	Multilabel accuracy Proposed	Multilabel accuracy MMSE
Arts	**0.233±0.011**✓	0.105±0.005	**0.423±0.01**✓	0.306±0.011	**0.405±0.01**✓	0.308±0.013
Education	**0.157±0.009**✓	0.105±0.003	**0.421±0.01**	0.349±0.01	**0.376±0.01**✓	0.322±0.009
Entertain	**0.266±0.015**✓	0.172±0.007	**0.442±0.011**✓	0.361±0.013	**0.397±0.008**✓	0.339±0.01
Health	**0.227±0.01**✓	0.125±0.007	**0.576±0.009**✓	0.464±0.01	**0.52±0.009**✓	0.418±0.012
Recreation	**0.324±0.012**✓	0.184±0.006	**0.441±0.01**✓	0.339±0.01	**0.411±0.01**✓	0.333±0.011
Reference	**0.131±0.006**✓	0.042±0.003	**0.45±0.014**✓	0.364±0.011	**0.446±0.014**✓	0.358±0.011
Science	**0.147±0.009**✓	0.047±0.003	**0.304±0.013**✓	0.208±0.01	**0.267±0.012**✓	0.21±0.011
Social	**0.153±0.01**✓	0.043±0.001	**0.532±0.008**✓	0.45±0.01	**0.544±0.009**✓	0.463±0.01
Society	**0.164±0.008**✓	0.084±0.005	**0.301±0.007**✓	0.254±0.008	**0.261±0.007**✓	0.247±0.007
Bibtex	**0.23±0.011**✓	0.154±0.005	**0.315±0.011**✓	0.247±0.01	**0.248±0.008**✓	0.197±0.003
Corel5k	**0.213±0.015**✓	0.013±0.001	**0.266±0.008**✓	0.09±0.005	**0.181±0.006**✓	0.06±0.007
Enron	**0.255±0.028**✓	0.117±0.008	**0.504±0.016**✓	0.385±0.012	**0.356±0.015**✓	0.267±0.009
Emotions	**0.642±0.031**✓	0.636±0.027	**0.677±0.029**✓	0.665±0.026	**0.569±0.031**✓	0.554±0.029
Slashdot	0.302±0.012	**0.32±0.01**✓	**0.57±0.016**✓	0.567±0.015	**0.554±0.017**✓	0.525±0.016
Avg. Rank.	**1.071**	1.929	**1**	2	**1**	2

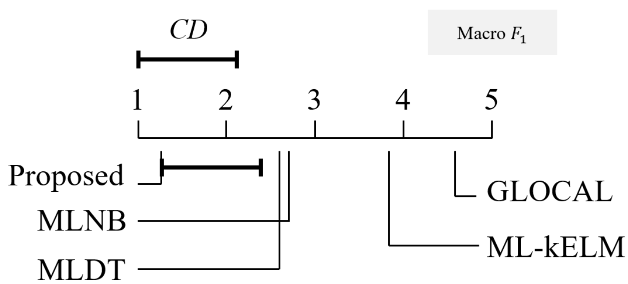

Fig. 1: Result of Bonferroni–Dunn test of Macro F_1

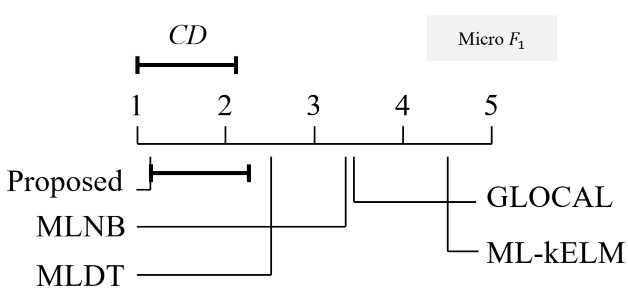

Fig. 2: Result of Bonferroni–Dunn test of Micro F_1

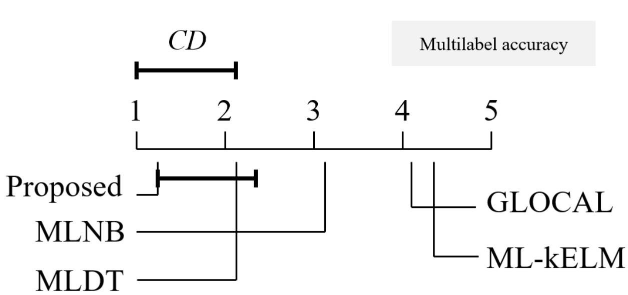

Fig. 3: Result of Bonferroni–Dunn test of Multilabel accuracy

for most multilabel datasets. Specifically, MLNB-LD achieves the highest performance on 86% of the datasets in terms of Macro F_1 and Micro F_1, and 79% of the datasets in terms of the multilabel accuracy. Consequently, MLNB-LD consistently achieves the highest average rank during all experiments. As shown in Fig. 1 and Fig. 2, MLNB-LD significantly outperforms MLNB, MLDT, ML-kELM, and GLOCAL in terms of Macro F_1 and Micro F_1. In addition, Fig. 3 show that MLNB-LD significantly outperforms MLDT, ML-kELM, and GLOCAL in terms of Macro F_1.

MLNB-LD uses the geometric mean to determine the final score, as shown in Eq. (7), instead of using a classical Bayesian estimation such as the minimum mean square error (MMSE) estimator, which may lead to a better classification performance. To verify this possibility, we conducted additional experiments by comparing the performances of two MLNB-LD variations with different aggregation processes: the geometric mean and an MMSE estimation (MMSE). Table 9 shows that MLNB-LD provides a significantly better classification performance than its counterpart for most of the datasets. In addition, we observed that both Friedman test and Bonferroni–Dunn test also confirmed the statistical superiority of MLNB-LD over MMSE. A possible reason for this result may be the sensitivity of the geometric mean regarding outlier values of $p(y_i)\prod_{j=1}^{m} p(x_j|y_i)\prod_{k=1}^{n} p(y_k|y_i)$ owing to the label sparsity of most of the multilabel dataset (Lee and Kim, 2016).

In a real-world situation, the multi-label classification problem may become more complicated by missing labels, indicating that the classifier may have to output label sets that are unobserved from the training process. To achieve this problem, MLNB-LD must be modified to consider all possible label sets instead of $\mathcal{Y}$. Although the computational cost can be increased exponentially owing to exhaustive multilabel learning setting, the classification performance may be varied. To show this aspect, we conducted the last experiments by comparing two variations with a different label set consideration; one is the label sets in $\mathcal{Y}$ (proposed), and the other is all possible label sets

Table 10: Comparison results of Proposed and EML on Emotions dataset

Evaluation measure	Proposed	EML
Macro F_1	**0.6412±0.0267**✓	0.6409±0.0271
Micro F_1	**0.6711±0.0258**✓	0.6708±0.0261
Multilabel Accuracy	**0.5609±0.0247**✓	0.5597±0.0252

(EML). Owing to the computational burden of the EML, we chose the Emotions dataset, which is composed of six labels. Thus, the EML must compute the possibility of $2^6 = 64$ label sets for each test instance despite there being only 27 distinct label sets in total. Table 10 summarizes the multilabel classification performance between MLNB-LD and EML in terms of three evaluation measures. The experimental results indicate that MLNB-LD can provide a similar multilabel classification performance without considering all possible label sets.

5. Conclusion

We presented a multilabel naïve Bayes classifier that considers the dependence among labels during classification. The proposed method utilizes the dependence between label pairs for determining the most probable label set for a given unseen instance. Our comprehensive experiments demonstrate that multilabel classification performance can be significantly improved by the proposed method. A comparison of the results obtained on 14 real-world datasets obtained from different domains shows the advantages of the proposed method compared with the four conventional multilabel classifiers in terms of three evaluation measures, i.e., Macro F_1, Micro F_1, and Multilabel accuracy. Thus, considering the dependence among labels is effective for solving the multilabel classification problem.

Future work should include the study of computational efficiency for utilizing label dependence in the multilabel classification process. In this study, the dependence between all label pairs is considered for identifying the most probable label set. This indicates that multilabel classification performance may be further improved if unnecessary or noisy information is removed. In addition, the experimental results demonstrate that the proposed method is computationally efficient because it identifies the most probable label set without considering all of the possible label sets. However, in the multilabel learning case in which the ground truth label set is partially given, the proposed method can be used to output the novel label sets by computing the score of the label sets that are unobserved from the training process. Furthermore, the proposed method uses the geometric mean for aggregating the score values obtained by conditioning each label. Although this demonstrates a superior multilabel classification performance, a different estimation or heuristic method can be considered to improve the multilabel classification performance. We intend to investigate this further in future work.

Acknowledgments

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (NRF-2019R1C1C1008404).

References

Cano, A., Luna, J.M., Gibaja, E.L., Ventura, S., 2016. LAIM discretization for multi-label data. Inf. Sci. 330, 370–384.

Demšar, J., 2006. Statistical Comparisons of Classifiers over Multiple Data Sets. J. Mach. Learn. Res. 7, 1–30.

Elghazel, H., Aussem, A., Gharroudi, O., Saadaoui, W., 2016. Ensemble multi-label text categorization based on rotation forest and latent semantic indexing. Expert Syst. Appl. 57, 1 – 11.

Huang, J., Li, G., Huang, Q., Wu, X., 2015. Learning label specific features for multi-label classification, in: Proc. 15th IEEE Int. Conf. Data Mining, Atlantic City, USA. pp. 181–190.

Huang, J., Li, G., Wang, S., Xue, Z., Huang, Q., 2017. Multi-label classification by exploiting local positive and negative pairwise label correlation. Neurocomputing 257, 164–174.

Jing, L., Shen, C., Yang, L., Yu, J., Ng, M.K., 2017. Multi-label classification by semi-supervised singular value decomposition. IEEE Trans. Image Process. 26, 4612–4625.

Kumar, V., Pujari, A.K., Padmanabhan, V., Sahu, S.K., Kagita, V.R., 2018. Multi-label Classification Using Hierarchical Embedding. Expert Syst. Appl. 91, 263–269.

Lee, J., Kim, D.W., 2016. Efficient multi-label feature selection using entropy-based label selection. Entropy 18, 40501–40526.

Lee, J., Seo, W., Park, J.H., Kim, D.W., 2019. Compact feature subset-based multi-label music categorization for mobile devices. Multimedia Tools Appl. 78, 4869–4883.

Li, X., Yang, B., 2018. A pseudo label based dataless naive bayes algorithm for text classification with seed words, in: Proc. 27th Int. Conf. Computational Linguistics, Santa Fe, USA. pp. 1908–1917.

Luo, F., Guo, W., Yu, Y., Chen, G., 2017. A multi-label classification algorithm based on kernel extreme learning machine. Neurocomputing 260, 313–320.

Nam, J., Mencía, E.L., Kim, H.J., Fürnkranz, J., 2017. Maximizing subset accuracy with recurrent neural networks in multi-label classification, in: Proc. 31th Ann. Conf. Neural Information Processing Systems, Long Beach, USA. pp. 5413–5423.

Read, J., Pfahringer, B., Holmes, G., Frank, E., 2011. Classifier chains for multi-label classification. Mach. Learn. 85, 333–359.

Trohidis, K., Tsoumakas, G., Kalliris, G., Vlahavas, I., 2011. Multi-label classification of music by emotion. EURASIP J. Audio Speech Music Process. 2011, 1–9.

Tsoumakas, G., Katakis, I., Vlahavas, I., 2010. Random k-labelsets for multilabel classification. IEEE Trans. Knowl. Data Eng. 23, 1079–1089.

Ueda, N., Saito, K., 2003. Parametric mixture models for multi-labeled text, in: Proc. 16th Ann. Conf. Neural Information Processing Systems, Vancouver, Canada. pp. 737–744.

Vens, C., Struyf, J., Schietgat, L., Džeroski, S., Blockeel, H., 2008. Decision trees for hierarchical multi-label classification. Mach. Learn. 73, 185.

Vluymans, S., Cornelis, C., Herrera, F., Saeys, Y., 2018. Multi-label classification using a fuzzy rough neighborhood consensus. Inf. Sci. 433, 96–114.

Wu, B., Lyu, S., Hu, B.G., Ji, Q., 2015. Multi-label learning with missing labels for image annotation and facial action unit recognition. Pattern Recognit. 48, 2279–2289.

Zhang, M.L., Peña, J.M., Robles, V., 2009. Feature selection for multi-label naïve bayes classification. Inf. Sci. 179, 3218–3229.

Zhang, M.L., Wu, L., 2015. LIFT: Multi-label learning with label-specific features. IEEE Trans. Pattern Anal. Mach. Intell. 37, 107–120.

Zhang, M.L., Zhou, Z.H., 2007. ML-KNN: A lazy learning approach to multi-label learning. Pattern Recognit. 40, 2038–2048.

Zhang, M.L., Zhou, Z.H., 2014. A review on multi-label learning algorithms. IEEE Trans. Knowl. Data Eng. 26, 1819–1837.

Zhu, Y., Kwok, J.T., Zhou, Z.H., 2018. Multi-label learning with global and local label correlation. IEEE Trans. Knowl. Data Eng. 30, 1081–1094.