Multimed Tools Appl
https://doi.org/10.1007/s11042-018-6100-8

Compact feature subset-based multi-label music categorization for mobile devices

Jaesung Lee[1] · Wangduk Seo[1] · Jin-Hyeong Park[1] · Dae-Won Kim[1]

Received: 28 November 2017 / Revised: 8 April 2018 / Accepted: 4 May 2018

Abstract Music categorization based on acoustic features extracted from music clips and user-defined tags forms the basis of recent music recommendation applications, because relevant tags can be automatically assigned based on the feature values and their relation to tags. In practice, especially for handheld lightweight mobile devices, there is a certain limitation on the computational capacity, owing to consumers' usage behavior or battery consumption. This also limits the maximum number of acoustic features to be extracted, and results in the necessity of identifying a compact feature subset that is used for the music categorization process. In this study, we propose an approach to compact feature subset-based multi-label music categorization for mobile music recommendation services. Experimental results using various multi-labeled music datasets reveal that the proposed approach yields better performance when compared to conventional approach.

Keywords Music information retrieval · Mobile devices · Multi-label learning · Hybrid search

1 Introduction

Recently, music recommendation applications, such as playlist recommendations [16], automated tagging systems [29], and emotion recognition [1], examined the popularity of social network services and smartphones [7]. One of the most important sub-tasks to realize a music recommendation service that can attract users is music categorization that identifies relevant tags or labels, such as genres and moods for each music clip. Additionally, studies considered applications in mobile situations because most users play music by using their handheld smartphones. Teng et al. [27] considered music recommendations based on

✉ Dae-Won Kim
dwkim@cau.ac.kr

[1] Chung-Ang University, Seoul, Republic of Korea

Published online: 11 May 2018

features gathered by using built-in sensors of smartphones. Kaminskas et al. [8] devised a music recommendation for a place of interest. Baltrunas et al. [2] proposed context-aware recommendations in car driving situations.

In conventional music categorization studies, hundreads of acoustic features including tempo, rhythm, and harmony, are extracted when a new music clip is detected, and subsequently a model that represents the relation between extracted feature values and tags is used to assign a set of relevant tags. However, especially with respect to applications that are executed from lightweight mobile devices, there is a maximum number of allowed features that are extracted due to low computational capacity [3, 20]. Thus, consumers may suffer from a low quality of user experience due to unacceptable waiting time or battery consumption if the model requires excessive acoustic features to identify relevant tags. This disadvantage can be overcame by building an accurate model that only depends on an acceptable number of acoustic features (or lower).

To achieve this, we propose an approach to compact feature subset-based multi-label music categorization. Our contribution is summarized as follows:

- We formulate this problem as a budgeted feature selection problem [19, 30]. However, to the best of the authors' knowledge, there is a paucity of studies that consider multi-label feature selection with budget constraints.
- We released 10 new datasets for music categorization studies that were obtained and validated through the national research projects from Korea.
- Our mathematical analysis indicates important features can be neglected during the search when the size of original feature set is excessively high when compared to the number of features to be selected.
- We demonstrate the performance of the state-of-the-art multi-label feature wrapper method on 12 multi-label music datasets.

2 Related work

In the music information retrieval community, modeling music categorization task as a multi-label learning problem attracted significant scientific interest [23]. In the works of [12, 13, 37], music emotion categorization is modeled as a multi-label classification because music datasets from music clips can be associated with multiple concurrent labels [13]. Further, in the study of [20], multi-label feature selection for music datasets that have a certain limitation on the maximum feature subset size is considered an important issue for the practical applications from mobile devices or for recommending music clips for users. Let $W \subset \mathbb{R}^d$ denotes a set of patterns constructed from a set of features F. In multi-label datasets, each pattern $w_i \in W$ where $1 \leq i \leq |W|$ is assigned to a certain label subset $\lambda_i \subseteq L$, where $L = \{l_1, \ldots, l_{|L|}\}$. The task of multi-label classification in multi-label datasets involves identifying a function that maps given instances into one of the $2^{|L|}$ label subsets based on input feature values. To enhance the performance of multi-label classification, a feature selection method is applied to determine a compact feature subset only contains relevant features [6, 9, 11, 13]. For this task, an evolutionary algorithm, namely the Genetic Algorithm (GA)-based multi-label feature selection method exhibits strength in terms of the classification performance [33].

In multi-label feature selection studies, one of the simplest method involves transforming label sets into a single label data, such as a label powerset, and applying the same conventional feature selection method as that of single label data [24, 25]. In addition to the advantages of the immediate use of conventional methods, they involve side effects [26] such as an imbalance in transformed single label data. Additionally, they commonly suffer from low multi-label classification performance due to the lack of interaction with multi-label classifiers. In the wrapper approach, Zhang et al. [35] proposed a multi-label feature selection method based on a GA that is the most frequent choice for evolutionary feature wrapper studies [28]. Specifically, the method combined instance- and label-based evaluation measures [34] as a fitness function to consider label dependency. However, in its original proposal, the maximum allowed number of features to be selected is not considered from the genetic search process. In the study of [12], the multi-label feature selection performance was demonstrated for comparison purposes while considering feature subset size. Specifically, each chromosome randomly selects features less than n in the initialization stage. During the GA search process, the budget constraint is continuously satisfied by employing restrictive operators [39].

Another method that considers the number of features to be selected is multi-objective evolutionary algorithm based feature selection [31, 36, 37]. This approach treats a reduction in the number of features for selection as another objective in the search process. Given more than one objective, a specifically-designed ranking method was proposed for a multi-objective problem, and this is termed as a non-dominated sort [5]. The rank of each chromosome is measured by the number of times the chromosome dominates other chromosomes in all objectives. The most common approach in multi-objective evolutionary algorithm is NSGA-II [17] although it was employed in other evolutionary search methods such as Particle Swarm Optimization (PSO) [37]. A common disadvantage of them is that solutions may not satisfy budget constraints.

3 Proposed approach

Conventional multi-label feature wrappers can fail to find an optimal solutions because they neglect important features during the genetic search process, and thereby resulting in the low performance of multi-label classification. We first demonstrate why an additional component is employed for multi-label feature selection with the budget constraint especially when $n \ll |F|$, and then explain our approach. A conventional GA is composed of four components, namely initialization, natural selection, genetic operation, and evaluation. Among them, new feature subsets are generated through initialization and genetic operation including crossover and mutation. However, conventional crossover mechanisms such as single-point, two-point, and uniform crossovers [18] do not introduce new features into the population because their algorithms involve two selected chromosomes producing offspring by swapping their parts between selected crossover points. Thus, if two ancestors do not activate important features then there is no probability of activation. Therefore, the GA must introduce new albeit important features by using the initialization and mutation. However, the algorithm naturally involves a low probability when the maximum number of feature subset is bounded to a low value.

Let n be the maximum allowed number of features that is selected from a set of features F where $|F| = d$. Each chromosome is allowed to randomly select n or lesser features in the initialization stage. Thus, if there is an important but unselected feature f that

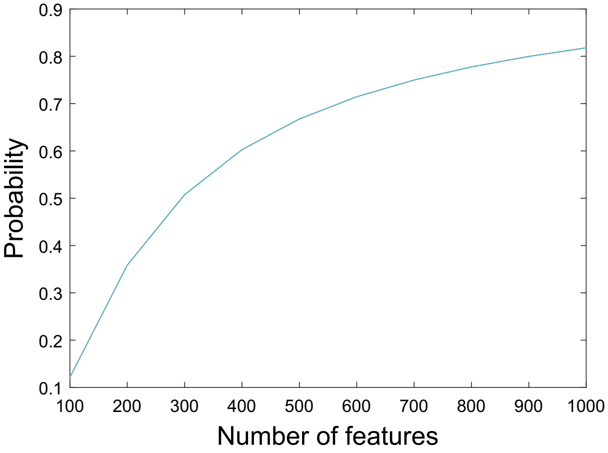


Fig. 1 Probability of neglecting an important feature in 20 chromosomes when maximum allowed number of features is limited to 10

significantly affects the performance of multi-label classification, then the probability x that a chromosome selects f is formulated as follows:

$$x = \sum_{k=1}^{n} \frac{1}{n} \cdot \frac{\binom{1}{1} \cdot \binom{N-1}{k-1}}{\binom{N}{k}} \tag{1}$$

The probability y that all m chromosomes are not selected f is as follows:

$$y = (1 - x)^m \tag{2}$$

Figure 1 shows the probability of not selecting an important feature f in 20 chromosomes when the maximum feature subset size is 10. If $d > 350$, then the probability y exceeds 50%. If $d \geq 1,000$, then the probability exceeds 80%. The figure indicates that it is difficult to introduce an important feature f into the current population without a specific procedure. Therefore, an efficient component should be employed to identify and introduce important features by considering their impact on the final multi-label categorization accuracy.

Algorithm 1 Compact feature subset selection

```
procedure FEATURE WRAPPER($v$, $m$)
    $t \leftarrow 0, u \leftarrow 0$                                      ▷ $t$-th generation
    initializing $P(t)$                                ▷ population $P$ of $t$-th generation
    evaluating $P(t)$
    while $u \leq v$ do                                ▷ if spent FFC $u$ is less than $v$
        create $N(t)$ using genetic operators and feature filter      ▷ offspring set $N(t)$
        evaluate $N(t)$ using a multi-label classifier
        add $N(t)$ to $P(t)$
        $t \leftarrow t + 1$
        select $P(t)$ from $P(t-1)$                                   ▷ natural selection
        $u \leftarrow m + |N(t)| \cdot t$                        ▷ update $u$ based on spent FFC
    end while
end procedure
```

Table 1 Notations used in the design of the proposed approach

Terms	Meanings
t	Number of generations
$P(t)$	The population at the t-th generation
m	The size of the population, $\|P(t)\| = m$
c	A chromosome in $P(t)$
S_c	A selected feature subset represented by c
$N(t)$	A set of newly created solutions from $P(t)$
v	Maximum number of allowed fitness function calls (FFCs)
u	Number of spent FFCs, $u = m + 2 \cdot \|G(t)\| \cdot t$
n	Maximum number of allowed features selected by S_c

Algorithm 1 outlines the procedures used in our approach. The terms used to describe the algorithm are summarized in Table 1. The representation of each chromosome is a binary string, and each bit represents an individual feature wherein those with values of one are selected and zero are unselected. In line 3 of the initialization stage, the algorithm generates m choromosomes by randomly selecting features less than n. Each chromosome is evaluated by using a fitness function, and we used a multi-label classification error to evaluate the fitness of selected chromosomes. To evaluate m chromosomes, and thus m Fitness Function Calls (FFCs) are spent from line 4. After performing the initialization process, offspring are reproduced. In our approach, the algorithm creates the offspring set $N(t)$ by using three mechanisms, namely restrictive crossover, restrictive mutation [38], and a conventional multi-label feature filter. The detailed procedure of the restrictive crossover and mutation can be described as follows:

- Restrictive crossover first selects two chromosomes $c1$ and $c2$ in $P(t)$. Let $k = min(|S_{c1}|, |S_{c2}|)$, where $|S_{c1}| \leq n$ and $|S_{c2}| \leq n$. Subsequently, two offspring are created by the swapping features selected by c_1 and c_2 for k times with the probability of 0.5.
- Restrictive mutation selects one chromosome $c1$. Subsequently, it creates a new chromosome by flipping the bit for the corresponding feature $|S_{c1}|$ times iteratively from selected to unselected, or unselected to selected. If $|S_{c1}|$ becomes larger than n, it chooses the $n - |S_{c1}|$ selected features and changes them to be unselected.

From the restrictive crossover and mutation, a set of features are randomly discarded to regulate the budget constraint if the offspring activates excessive features. However, the analysis indicated that important features are missed if the algorithm only exploits the conventional genetic operators. In order to achieve this, a state-of-the-art multi-label feature filter is employed to introduce unselected albeit important features into the current population [13]. Let S_c be a feature subset that is selected by a chromosome $c \in N(t)$. With respect to an unselected feature subset $\{F - S_c\}$, the algorithm creates new chromosomes by iteratively selecting a feature f_i that satisfies the following criterion until a new chromosome contains $|S_c|$ features to regulate the budget constraint. This is expressed as follows:

$$\max_{f_i \in \{F - S_c\}} \left[\sum_{l \in L} M(f_i; l) - \sum_{f \in S_{i-1}} \sum_{l \in L} \frac{M(f_i; l)}{H(f_i)} M(f_i; f) \right] \quad (3)$$

where $M(x;y) = H(x) - H(x,y) + H(y)$ denotes the mutual information between variables x and y, and $H(x) = -\sum P(x) \log P(x)$ denotes the joint entropy with their probability functions $P(x)$, $P(y)$, and $P(x,y)$. The algorithm repeats additional offspring creation with respect to each chromosome created by genetic operators to ensure a balance between a stochastic global search and local search [12].

To explain how the employed feature filter effectively introduces the novel features into the population, we demonstrate a step-by-step computation of the feature filter using a toy dataset presented in Table 2. The dataset is composed of eight patterns, six features $\{f_a, \ldots, f_f\}$, and two labels $\{l_a, l_b\}$. Although we created the toy dataset using categorical features for simplicity. Numeric features are also included in the dataset. For such cases, the entropy calculation for scoring the feature importance can be performed using the probability density function [22] or after discretization of numeric features [4]. Suppose a chromosome activates two features f_b and f_f; then, $S_c = \{f_b, f_f\}$. The goal of the employed feature filter is to identify the important features from $\{F - S_c\} = \{f_a, f_c, f_d, f_e\}$ for creating a new feature subset S_e, where $|S_e| = |S_c|$. Starting from $S_0 = \{\emptyset\}$, the feature filter should calculate the mutual information value of each feature in $\{F - S_c\}$. Here, we focus on the calculation of f_a. This can be calculated as

$$\sum_{l \in \{l_a, l_b\}} M(f_a; l) - \underbrace{\sum_{f \in \{\emptyset\}} \sum_{l \in \{l_a, l_b\}} \frac{M(f_a; l)}{H(f_a)} M(f_a; f)}_{\text{Part 1}}$$

Because $S_0 = \{\emptyset\}$, Part 1 can be canceled. Thus, the computation can be simplified as

$$\begin{aligned}\textstyle\sum_{l \in \{l_a, l_b\}} M(f_a; l) &= M(f_a; l_a) + M(f_a; l_b) \\ &= H(f_a) - H(f_a, l_a) + H(l_a) + H(f_a) - H(f_a, l_b) + H(l_b)\end{aligned}$$

The joint entropy $H(f_a, l_a)$ is calculated as

$$\begin{aligned}H(f_a, l_a) &= -\textstyle\sum_{x \in \{f_a, l_a\}} P(x) \log P(x) \\ &= -P(f_a = \text{a}1, l_a = 0) \log P(f_a = \text{a}1, l_a = 0) \\ &\quad -P(f_a = \text{a}2, l_a = 0) \log P(f_a = \text{a}2, l_a = 0) \\ &\quad -P(f_a = \text{a}1, l_a = 1) \log P(f_a = \text{a}1, l_a = 1) \\ &\quad -P(f_a = \text{a}2, l_a = 1) \log P(f_a = \text{a}2, l_a = 1) \\ &= -\tfrac{2}{8} \log \tfrac{2}{8} - \tfrac{2}{8} \log \tfrac{2}{8} - \tfrac{2}{8} \log \tfrac{2}{8} - \tfrac{2}{8} \log \tfrac{2}{8} = 2.0\end{aligned}$$

Table 2 A toy dataset for demonstration

Features						Labels	
f_a	f_b	f_c	f_d	f_e	f_f	l_a	l_b
a1	b2	c1	d1	e1	f2	0	1
a2	b1	c1	d1	e1	f2	0	1
a1	b1	c2	d1	e1	f1	0	0
a2	b1	c2	d2	e1	f2	0	0
a1	b1	c1	d2	e2	f2	1	0
a2	b1	c1	d2	e2	f2	1	0
a1	b1	c2	d2	e2	f2	1	0
a2	b1	c2	d1	e2	f2	1	0

Similarly, other entropy terms can also be calculated where $H(f_a) = 1.0$, $H(l_a) = 1.0$, $H(f_a, l_b) = 1.8$ and $H(l_b) = 0.8$. Because all the entropy terms that are required for calculating the importance of f_a are in the first iteration, the algorithm is now able to evaluate the importance of f_a. By applying the same procedure, the importance values of features in $\{F - S_c\}$ can also be calculated, where the exact values of importance score for f_a is 0.0, f_c is 0.3, f_d is 0.5 and f_e is 1.3. Consequently, f_e will be selected at the first iteration owing to the highest importance score among f_a, f_c, f_d, and f_e. Next, because $|S_c| = 2$, the algorithm continues to identify the best feature among f_a, f_c, and f_d when f_e is already selected. Considering f_a for example, the algorithm evaluates the importance score of f_a as

$$\begin{aligned} &\textstyle\sum_{l\in\{l_a,l_b\}} M(f_a;l) - \sum_{f\in\{f_e\}} \sum_{l\in\{l_a,l_b\}} \frac{M(f_a;l)}{H(f_a)} M(f_a;f) \\ &= M(f_a;l_a) + M(f_a;l_b) - \tfrac{M(f_a;l_a)}{H(f_a)} M(f_a;f_e) - \tfrac{M(f_a;l_b)}{H(f_a)} M(f_a;f_e) \end{aligned}$$

Similarly, the importance values of other features, f_c and f_d can also be calculated where the exact values of importance score for f_a is 0.0, f_c is 0.3, and f_d is 0.4. Thus, f_d will be selected at the second iteration owing to the highest importance score among f_a, f_c, and f_d. Subsequently, the filter will set $S_e = \{f_d, f_e\}$ and a chromosome that activates S_e is created accordingly. This newly created chromosome is included in the offspring set $N(t)$. It should be noted that different multi-label feature filters can be employed for this purpose although we select this filter with respect to its robustness to the number of labels that varies according to the application. Next, the algorithm spends $|N(t)|$ of FFCs to evaluate the performance of chromosomes in $N(t)$ (line 7). Subsequently, $N(t)$ is added to $P(t)$ (line 8) and m chromosomes with higher fitness values are selected (line 10). The procedure is terminated when FFCs spent exceeds allowed FFCs, denoted as v, parameter given by users.

Finally, we conducted a formal analysis on the computational cost of the proposed method. Let $\alpha(S_c)$ be the computation time required to evaluate feature subset S_c, where $|S_c| \ll n$; S_c could contain a maximum of n features in the worst case. The value of $\alpha(S_c)$ can be influenced by the size of the training patterns and the number of given labels; however, it can be regarded as a constant value because the size of the training patterns and the number of given labels do not change during the training process. Consequently, the major factor of $\alpha(S_c)$ is the size of S_c. The evaluation of a single feature through the spending of an FFC is called a basic operation, and $\alpha(1)$ for a basic operation is referred to as the basic time. For a simpler analysis, we assumed that $\alpha(S_c) \approx |S_c| \times \sigma$, where σ represents the basic time. Based on these notations, we analyzed the computational complexity of the proposed method. During the initialization step, the algorithm creates m chromosomes and evaluates them. Because each solution is allowed to contain a maximum of n features, $O(mn\sigma)$ computations must be spent for the initialization step. Subsequently, $|N(t)|$ chromosomes are created using genetic operators and evaluated using the fitness function, spending $O(|N(t)|n\sigma$. In the feature filter process for identifying novel unselected features, $|N(t)|$ chromosomes conveying novel features will be created and evaluated again. This also consumes $O(|N(t)|n\sigma)$ computations because each chromosome only activates a maximum of $|S_c| \leq n$ features. Therefore, the computational cost of each generation is $O(2|N(t)|n\sigma)$. Because the maximum number of generations is approximately $(v-m)/2|N(t)|$, we obtain $(v-m)n\sigma$ by multiplying $(v-m)/2|N(t)|$ with $2|N(t)|n\sigma$. The computational cost of the proposed method is therefore $O(mn\sigma + (v-m)n\sigma) = O(vn\sigma)$. Because σ is constant, our analysis indicates that the computational cost is influenced by the maximum number of allowed FFCs v, and the maximum number of allowed features n. The same result can be obtained for conventional evolutionary algorithms employed for the performance comparison in the next section.

4 Experimental results

We experimented with 12 music datasets that were collected through a national research project in Korea with the exception of CAL500 and Emotions. The CAL500 and Emotions datasets are generated from a music tag annotation application in which the music retrieval system learns a relation between acoustic features and words from a dataset of annotated audio tracks [21]. The remaining ten datasets were not publicized in the music information retrieval community, and thus we summarize their details as follows:

- Audionautix: Audionautix dataset contains 308 non-vocal music clips composed for the background music of games, and each music clip is assigned to 68 music genres annotated by its composer. The acoustic features are extracted by the MIR toolbox [10]. Unnecessary features are discarded such as features with zero variance.
- Bugs2664 and BugsEmo: Bugs2664 dataset is created by gathering 2,664 music clips from an online music streaming service in Korea, and most of them correspond to K-pop music. Specifically, each music clip is assigned to 40 musical tags that are categorized into seasons, emotions, usage, and places. BugsEmo dataset is created by subsampling a Bugs2664 dataset by only considering seven emotion tags. The acoustic features of these two datasets are extracted by the MIR toolbox.
- China3004: China3004 dataset contains 3,004 music clips gathered from online music websites in China, and each music clip is assigned to a set of 28 musical styles provided by an online system. The acoustic features of the China3004 dataset are extracted by the MIR toolbox.
- Style812, Genre3, and Highlight: In the Style812 dataset, 812 music clips are labeled to three music styles including Rhythmic, Romantic, and Melancholy. The Genre3 dataset is created by extracting acoustic features from 812 music clips that correspond to the same music clips in the Style812 dataset. The dataset is created to identify time-variant musical themes including genres, highlights, or emotions. Thus, it contains all the music pieces as opposed to simply selecting a representative piece for each music clip. Similarly, the Highlight dataset is created by using the same procedure as that for Genre3 although the label indicates that each piece is the highlight of corresponding music clips. The acoustic features of all three datasets are extracted by the MIR toolbox.
- KOCCA40: This dataset is created for the education of music information retrieval from the undergraduate classes. In order to encourage students, 40 music clips that are confirmed as easily trained by a machine learning algorithm are selected. Each music clip is assigned to four different labels Passionate, Breezy, Depressed, and Peaceful. The acoustic features of the KOCCA40 dataset are extracted by the MIR toolbox.
- MusicEmo-A and MusicEmo-B: In MusicEmo-A, 864 acoustic features are extracted by the MIR toolbox from 100 collected music clips and approximately labeled 500 times through an on-line annotation system. In MusicEmo-B, 346 audio features are extracted by the MIR toolbox from 565 music clips and labeled approximately 3600 times. Each music clip is assigned to relevant tags including Excitement, Distress, Depression, and Contentment. Earlier versions of these two datasets has been publicized from our previous study [14]. In the current study, 21 errors of feature values are corrected.

The acoustic features of all datasets are extracted from music chunks starting from 0 s and ending at 40 s, which can be categorized into dynamics, fluctuation, rhythm, spectral, and tonal features. All the datasets are able to download from our website: http://mi.cau.ac.kr/?f=teaching&m=prog_amc.

Table 3 lists the standard statistics of the music multi-label datasets employed in our experiments where $|W|$ denotes the number of patterns in the dataset, $|F|$ denotes the number of features, and $|L|$ denotes the number of labels. The label cardinality *Card* represents the average number of labels for each instance. The label density *Den* denotes the label cardinality with respect to the total number of labels. The number of distinct label sets *Distinct* indicates the number of unique label subsets in L. *Subject* represents the applications that the label of each dataset is related to. Our problem is that the maximum feature subset size is bounded, and thus we compared the proposed approach with GA with restrictive genetic operators [38] (RGA) which satisfies the problem constraints. The proposed approach and the RGA's maximum feature subset size are set as 10. In order to evaluate the quality of a feature subset obtained through each method, we considered a conventional multi-label classifiers, namely the Multi-label Naive Bayes (MLNB) classifier [35]. For fairness, we conducted a holdout cross-validation for each experiment [12]; 80% of the patterns in a given dataset were randomly chosen as the training set for multi-label feature selection and classifier training, and the remaining 20% of the patterns was used as the test set to obtain the multi-label classification performance that is reported. With respect to the RGA, and the proposed approach, we set the size of population as 20, and the maximum number of allowed fitness function calls was set as 100. Each experiment was repeated 10 times, and the average value was used to represent the classification performance according to each feature selection method.

To measure the performance, we employed the following four evaluation measures: Hamming loss, Multi-label accuracy, Ranking loss, and Normalized coverage. It is assumed that $T = \{(T_i, \lambda_i)|1 \leq i \leq |T|\}$ is a given test set where $\lambda_i \subseteq L$ denotes a correct label subset. According to a test pattern T_i, a classifier, such as MLNB, outputs a set of confidence values $0 \leq \psi_{i,l} \leq 1$ for each label $l \in L$. If confidence value $\psi_{i,l}$ exceeds a predefined threshold value, such as 0.5, then the corresponding label l is included in the predicted label subset Y_i. Based on the ground truth λ_i, confidence values $\psi_{i,l}$, and predicted label

Table 3 Standard characteristics of the employed datasets

Dataset	$\|W\|$	$\|F\|$	$\|L\|$	Card.	Den.	Distinct.	Subject
Audionautix	308	376	68	4.731	0.070	254	Genre
Bugs2664	2664	137	40	1.917	0.048	666	Tag
BugsEmo	753	109	7	1.000	0.143	7	Emotion
CAL500	502	68	174	26.044	0.150	502	Tag
China3004	3004	345	28	1.000	0.036	28	Style
Emotions	593	72	6	1.868	0.311	27	Emotion
Genre3	2597	365	3	1.000	0.333	3	Genre
Highlight	2597	365	2	1.000	0.500	2	Highlight
KOCCA40	40	123	4	1.000	0.250	4	Emotion
MusicEmo-A	100	864	4	1.530	0.383	11	Emotion
MusicEmo-B	565	346	4	1.292	0.323	9	Emotion
Style812	812	348	3	1.000	0.333	3	Style

subset Y_i, the multi-label classification performance is measured according to each evaluation measure [15, 25, 32]. Hamming loss is defined as follows:

$$hloss(T) = \frac{1}{|T|}\sum_{i=1}^{|T|}\frac{1}{|L|}|\lambda_i \triangle Y_i|$$

where $\triangle$ denotes the symmetric difference between two sets. Multi-label accuracy is defined as follows:

$$mlacc(T) = \frac{1}{|T|}\sum_{i=1}^{|T|}\frac{|\lambda_i \cap Y_i|}{|\lambda_i \cup Y_i|}$$

Ranking loss is defined as follows:

$$rloss(T) = \frac{1}{|T|}\sum_{i=1}^{|T|}\frac{\left|\{(a,b)|a \in \lambda_i, b \in \overline{\lambda_i}, \psi_{i,a} \leq \psi_{i,b}\}\right|}{|\lambda_i||\overline{\lambda_i}|}$$

where $\overline{\lambda_i}$ denotes a complementary set of λ_i. Thus, Ranking loss measures the average fraction of (a, b) pairs with respect to $\psi_{i,a} \leq \psi_{i,b}$ against all possible relevant and irrelevant label pairs. Normalized coverage is defined as follows:

$$ncov(T) = \frac{1}{|L|}\left(\frac{1}{|T|}\sum_{i=1}^{|T|}\max_{l\in\lambda_i} rank(l) - 1\right)$$

where $rank(\cdot)$ returns the rank of the corresponding relevant label $l \in \lambda_i$ according to $\psi_{i,l}$ in non-increasing order. Thus, Normalized coverage measures the number of labels that should be determined as positive for all the relevant labels to be positive. Lower values of Hamming loss, Ranking loss, and Normalized coverage, and higher value of Multi-label accuracy indicate good classification performance.

Tables 4 and 5 list the experimental results for the proposed approach and RGA on 12 multi-label datasets, presented as the average performances for holdout cross-validation with corresponding standard deviations. Table 4 contains the performance results for multi-label accuracy and hamming loss, and Table 5 contains the performance results for ranking loss and normalized coverage. The better performance between the two approaches is indicated with a bold font and a ✓symbol. Based on the experimental results, we summarize our observations as follows:

- In terms of the Multi-label accuracy performance, the proposed approach exhibited better performances with the exception of KOCCA40 dataset.
- In terms of the Hamming loss performance, the proposed approach achieved better performances for BugsEmo, CAL500, Emotions, Genre3, KOCCA40, MusicEmo-A, MusicEmo-B, and Style812 datasets.
- In terms of the ranking loss performance, the proposed approach exhibited better performances than conventional approach for nine datasets. Excpetions are Audionautix, CAL500, and Highlight datasets.
- In terms of the normalized coverage performance, better performances were achieved by the proposed approach for eight datasets. Exceptions are Bugs2664, BugsEmo, CAL500, and Highlight datasets.
- In the experiments of Emotions, Genre3, MusicEmo-A, MusicEmo-B, and Style812 datasets, the proposed approach always yields better performances with respect to all the evaluation measures.

Table 4 Comparison results for multi-label feature selection methods in terms of the multi-label accuracy and Hamming loss (mean ± std. deviation)

Evaluation measure	Multi-label accuracy		Hamming loss	
Methods	Proposed	RGA	Proposed	RGA
Audionautix	**0.190 ± 0.025**✓	0.189 ± 0.034	0.070 ± 0.004	**0.070 ± 0.003**✓
Bugs2664	**0.053 ± 0.013**✓	0.031 ± 0.005	0.048 ± 0.001	**0.048 ± 0.001**✓
BugsEmo	**0.371 ± 0.040**✓	0.323 ± 0.044	**0.126 ± 0.006**✓	0.127 ± 0.008
CAL500	**0.193 ± 0.012**✓	0.192 ± 0.011	**0.137 ± 0.003**✓	0.137 ± 0.003
China3004	**0.063 ± 0.014**✓	0.017 ± 0.007	0.036 ± 0.000	**0.035 ± 0.000**✓
Emotions	**0.480 ± 0.046**✓	0.475 ± 0.041	**0.229 ± 0.024**✓	0.245 ± 0.014
Genre3	**0.917 ± 0.013**✓	0.912 ± 0.018	**0.041 ± 0.008**✓	0.058 ± 0.010
Highlight	**0.735 ± 0.017**✓	0.734 ± 0.017	0.266 ± 0.017	0.266 ± 0.017
KOCCA40	0.588 ± 0.156	**0.656 ± 0.165**✓	**0.150 ± 0.084**✓	0.169 ± 0.075
MusicEmo-A	**0.624 ± 0.058**✓	0.514 ± 0.082	**0.245 ± 0.056**✓	0.264 ± 0.048
MusicEmo-B	**0.568 ± 0.064**✓	0.567 ± 0.052	**0.216 ± 0.028**✓	0.219 ± 0.031
Style812	**0.826 ± 0.017**✓	0.812 ± 0.031	**0.097 ± 0.016**✓	0.111 ± 0.023

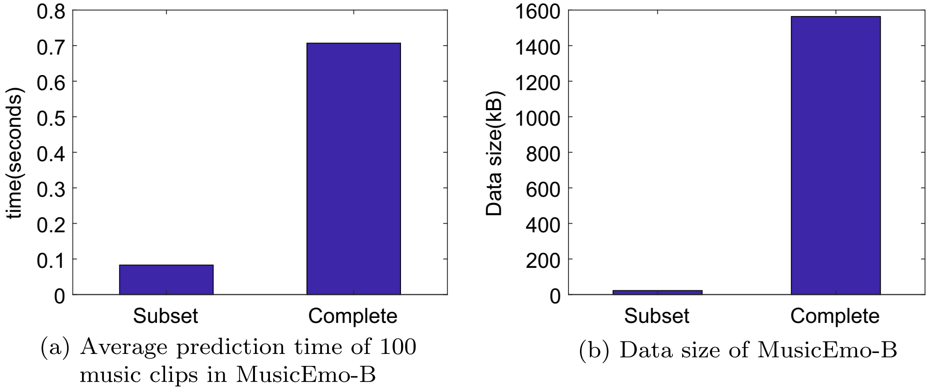
The ✓ symbol indicates the method that achieves better performance for each dataset

Finally, to indicate the efficiency of our proposed approach for mobile devices, we checked the average prediction time and data size for prediction for the MusicEmo-B dataset. In Fig. 2a, the plot indicates the average prediction time for 100 music clips, and in Fig. 2b, the plot indicates the data size that was used for training and prediction. We compared these parameters for the complete feature set, shown as "Complete" in both plots, and the feature subset, where the number of employed features by the model obtained by

Table 5 Comparison results for multi-label feature selection methods in terms of the ranking loss and normalized coverage (mean ± std. deviation)

Evaluation measure	Ranking loss		Normalized coverage	
Methods	Proposed	RGA	Proposed	RGA
Audionautix	0.207 ± 0.016	**0.207 ± 0.015**✓	**0.472 ± 0.026**✓	0.474 ± 0.019
Bugs2664	**0.260 ± 0.006**✓	0.276 ± 0.015	0.353 ± 0.010	**0.313 ± 0.009**✓
BugsEmo	**0.196 ± 0.016**✓	0.207 ± 0.013	0.314 ± 0.015	**0.313 ± 0.009**✓
CAL500	0.179 ± 0.006	**0.177 ± 0.005**✓	0.748 ± 0.013	**0.746 ± 0.013**✓
China3004	**0.217 ± 0.007**✓	0.226 ± 0.010	**0.245 ± 0.007**✓	0.254 ± 0.009
Emotions	**0.198 ± 0.032**✓	0.215 ± 0.022	**0.493 ± 0.024**✓	0.501 ± 0.026
Genre3	**0.026 ± 0.007**✓	0.039 ± 0.012	**0.351 ± 0.004**✓	0.360 ± 0.005
Highlight	0.266 ± 0.017	0.266 ± 0.017	0.633 ± 0.009	0.633 ± 0.009
KOCCA40	**0.113 ± 0.074**✓	0.117 ± 0.136	**0.356 ± 0.063**✓	0.375 ± 0.072
MusicEmo-A	**0.123 ± 0.030**✓	0.221 ± 0.052	**0.511 ± 0.067**✓	0.558 ± 0.074
MusicEmo-B	**0.175 ± 0.027**✓	0.196 ± 0.035	**0.458 ± 0.028**✓	0.483 ± 0.030
Style812	**0.074 ± 0.015**✓	0.099 ± 0.021	**0.382 ± 0.010**✓	0.397 ± 0.018

The ✓ symbol indicates the method that achieves better performance for each dataset



(a) Average prediction time of 100 music clips in MusicEmo-B

(b) Data size of MusicEmo-B

Fig. 2 Comparison results of the average prediction time and data size between the proposed approach and the conventional complete feature set-based approach (Average multi-label accuracy: 0.565 and 0.514, respectively)

proposed approach is 5 and is shown as "Subset" in both plots. In Fig. 2a, we checked the prediction time for 100 music clips in 100 iterations in MATLAB, and calculated the average. It took 0.083 seconds for predicting 100 music clips using the selected feature subset, which is approximately 8.5 times faster than that using the complete feature set. In Fig. 2b, we measured the data size for training and prediction. Using the feature subset, 22.60 kB of memory was required for training and prediction, which is better than 1563.92 kB when using the complete feature subset. Futhermore, the average multi-label accuracy of the feature subset and the complete feature set was 0.565 and 0.514, respectively, which indicates that the performance of the feature subset is better, and thus, indicates the efficiency of proposed approach.

5 Conclusion

In this study, we proposed a compact feature subset-based multi-label music categorization approach for mobile music recommendation services. To circumvent unacceptable power consumption and memory problem while satisfying consumers' interest, a model that only depends on a few acoustic features was considered. Our experiments on 12 real-world music multi-label datasets demonstrated that the proposed approach is more suitable than the conventional approach with respect to various measures.

Among 12 datasets, the cardinality of six datasets such as BugsEmo, China3004, Genre3, Highlight, and KOCCA40 is commonly one, indicating that they can also be regarded as the datasets of the conventional single-label classification problem, i.e., each music clip is assigned to only one label. Thus, the classification performance can be improved by allowing the classifier to select only one label for each unseen pattern. In fact, we have observed the improvement in accuracy by 15.37% from the additional experiment using the conventional naive Bayes classifier on the China3004 dataset after transforming it into a single-label dataset. This means that the additional information on the label space can be beneficial; the labels are mutually exclusive. In addition, future research may also include improvements to the categorization accuracy by obtaining a better hybridization of evolutionary search and multi-label feature filter because many evolutionary algorithms and multi-label feature filter methods are potential candidates for our algorithm. These additions and improvements will lead to different results. Therefore, we would like to further study this issue.

Acknowledgements This research was supported by the Chung-Ang University Research Scholarship Grants in 2018 and by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (NRF-2016R1C1B1014774).

References

1. Bai J, Feng L, Peng J, Shi J, Luo K, Li Z, Liao L, Wang Y (2016) Dimensional music emotion recognition by machine learning. Int J Cogn Inf Nat Intell 10(4):74–89
2. Baltrunas L, Kaminskas M, Ludwig B, Moling O, Ricci F, Aydin A, Lüke K-H, Schwaiger R (2011) Incarmusic: context-aware music recommendations in a car. In: Proceedings of the 12th international conference on electronic commerce and web technologies. Toulouse, pp 89-100
3. Blume H, Bischl B, Botteck M, Igel C, Martin R, Roetter G, Rudolph G, Theimer W, Vatolkin I, Weihs C (2011) Huge music archives on mobile devices. IEEE Signal Process Mag 28(4):24–39
4. Cano A, Luna JM, Gibaja EL, Ventura S (2016) LAIM discretization for multi-label data. Inform Sci 330(1):370–384
5. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6(2):182–197
6. Doquire G, Verleysen M (2013) Mutual information-based feature selection for multilabel classification. Neurocomputing 122(1):148–155
7. Fu Z, Lu G, Ting KM, Zhang D (2011) A survey of audio-based music classification and annotation. IEEE Trans Multimed 13(2):303–319
8. Kaminskas M, Ricci F (2011) Location-adapted music recommendation using tags. In: Proceedings of the 19th international conference on user modeling, adaptation, and personalization. Girona, pp 183-194
9. Kong D, Ding C, Huang H, Zhao H (2012) Multi-label ReliefF and F-statistic feature selections for image annotation. In: Proceeding of IEEE Conference on computer vision and pattern recognition. Providence, pp 2352–2359
10. Lartillot O, Toiviainen P (2007) A matlab toolbox for musical feature extraction from audio. In: Proceedings of the 10th International conference on digital audio effects. Bordeaux, pp 237–244
11. Lee J, Kim D-W (2015) Fast multi-label feature selection based on information-theoretic feature ranking. Pattern Recogn 48(9):2761–2771
12. Lee J, Kim D-W (2015) Memetic feature selection algorithm for multi-label classification. Inform Sci 293(1):80–96
13. Lee J, Kim D-W (2017) SCLS: multi-label feature selection based on scalable criterion for large label set. Pattern Recogn 66(1):342–352
14. Lee J, Jo J-H, Lim H, Chae J-H, Lee S-U, Kim D-W (2015) Investigating relation of music data: emotion and audio signals. Lect Notes Electr Eng 330(1):251–256
15. Lee J, Kim H, Kim N-R, Lee J-H (2016) An approach for multi-label classification by directed acyclic graph with label correlation maximization. Inform Sci 351(1):101–114
16. Liebman E, Saar-Tsechansky M, Stone P (2015) Dj-mc: a reinforcement-learning agent for music playlist recommendation. In: Proceedings of the 2015 International conference on autonomous agents and multiagent systems. IStanbul, pp 591–599
17. Lin Y, Hu Q, Liu J, Duan J (2015) Multi-label feature selection based on max-dependency and min-redundancy. Neurocomputing 168(1):92–103
18. Magalhaes-Mendes J (2013) A comparative study of crossover operators for genetic algorithms to solve the job shop scheduling problem. WSEAS Trans Comput 12(4):164–173
19. Min F, Xu J (2016) Semi-greedy heuristics for feature selection with test cost constraints. Granular Comput 1(3):199–211
20. Naula P, Airola A, Salakoski T, Pahikkala T (2014) Multi-label learning under feature extraction budgets. Pattern Recogn Lett 40(1):56–65
21. Ness SR, Theocharis A, Tzanetakis G, Martins LG (2009) Improving automatic music tag annotation using stacked generalization of probabilistic SVM outputs. In: Proceedings of the 17th ACM international conference on multimedia. Beijing, pp 705–708
22. Nguyen HB, Xue B, Andreae P (2016) Mutual information for feature selection: estimation or counting? Evol Intel 9(3):95–110

23. Papanikolaou Y, Katakis I, Tsoumakas G (2016) Hierarchical partitioning of the output space in multi-label data arXiv:1612.06083
24. Read J (2008) A pruned problem transformation method for multi-label classification. In: Proceedings of New Zealand computer science research student conference. Christchurch, pp 143–150
25. Spolaôr N, Monard MC, Tsoumakas G, Lee HD (2016) A systematic review of multi-label feature selection and a new method based on label construction. Neurocomputing 180(1):3–15
26. Sun Y, Wong A, Kamel M (2009) Classification of imbalanced data: a review International. J Pattern Recogn Artif Intell 23(4):687–719
27. Teng Y-C, Kuo Y-S, Yang Y-H (2013) A large in-situ dataset for context-aware music recommendation on smartphones. In: Proceedings of the 2013 IEEE international conference on multimedia and expo workshops. San Jose, pp 1–4
28. Xue B, Zhang M, Browne WN, Yao X (2016) A survey on evolutionary computation approaches to feature selection. IEEE Trans Evol Comput 20(4):606–626
29. Yan Q, Ding C, Yin J, Lv Y (2015) Improving music auto-tagging with trigger-based context model. In: Proceedings of the 2015 IEEE international conference on acoustics, speech and signal processing. Brisbane, pp 434–438
30. Yang H, Xu Z, Lyu MR, King I (2015) Budget constrained non-monotonic feature selection. Neural Netw 71(1):214–224
31. Yin J, Tao T, Xu J (2015) A multi-label feature selection algorithm based on multi-objective optimization. In: Proceedings of the 2015 International joint conference on neural networks. Killarney, pp 1–7
32. Zhang M-L, Wu L (2015) LIFT: multi-label learning with label-specific features. IEEE Trans Pattern Anal Mach Intell 37(1):107–120
33. Zhang M-L, Zhou Z-H (2007) ML-kNN: a lazy learning approach to multi-label learning. Pattern Recogn 40(7):2038–2048
34. Zhang M-L, Zhou Z-H (2014) A review on multi-label learning algorithms. IEEE Trans Knowl Data Eng 26(8):1819–1837
35. Zhang M-L, Peña JM, Robles V (2009) Feature selection for multi-label naive Bayes classification. Inform Sci 179(19):3218–3229
36. Zhang Y, Gong D-W, Rong M (2015) Multi-objective differential evolution algorithm for multi-label feature selection in classification. Lect Notes Comput Sci 9140(1):339–345
37. Zhang Y, Gong D-W, Sun X-Y, Guo Y-N (2017) A PSO-based multi-objective multi-label feature selection method in classification. Sci Rep 7(376):1–12
38. Zhu Z, Ong Y-S, Dash M (2007) Wrapper-filter feature selection algorithm using a memetic framework. IEEE Int Conf Syst Man Cybern Part B 37(1):70–76
39. Zhu Z, Jia S, Ji Z (2010) Towards a memetic feature selection paradigm. IEEE Comput Intell Mag 5(2):41–53

Jaesung Lee is currently an assistant professor in the School of Computer Science and Engineering, Chung-Ang Univ. in Seoul, Korea. Prior to coming to CAU, he did his postdoc, Ph.D., M.S. and B.S. at Chung-Ang Univ., Korea. His research interest includes advanced machine learning algorithms with innovative applications to music emotion recognition, euducational data mining, affective computing, and robot interaction.

Wangduk Seo is currently a Master student at Chung-Ang Univ. in Seoul, Korea, in the school of computer science and engineering, which he joinded in 2017.

Jin-Hyeong Park is currently a Master student at Chung-Ang Univ. in Seoul, Korea, in the school of computer science and engineering, which he joinded in 2018.

Dae-Won Kim is currently a professor in the School of Computer Science and Engineering, Chung-Ang University in Seoul, Korea. Prior to coming to Chung-Ang University, he did his postdoc, Ph.D., M.S. at KAIST, and the B.S. at Kyungpook National University, Korea. His research interest includes advanced data mining algorithms with innovative applications to bioinformatics, music emotion recognition, educational data mining, affective computing, and robot interaction.