SMART: Semi-Supervised Music Emotion Recognition with Social Tagging

Bin Wu*, Erheng Zhong*, Derek Hao Hu*, Andrew Horner*, Qiang Yang*

Abstract

Music emotion recognition (MER) aims to recognize the affective content of a piece of music, which is important for applications such as automatic soundtrack generation and music recommendation. MER is commonly formulated as a supervised learning problem. In practice, except for Pop music, there is little labeled data in most genres. In addition, emotion is genre specific in music and thus the labeled data of Pop music cannot be used for other genres. In this paper, we aim to solve the genre-specific MER problem by exploiting two kinds of auxiliary data: *unlabeled songs* and *social tags*. However, using these two kinds of data effectively is a non-trivial task, e.g. tags are noisy and therefore cannot be treated as fully trustworthy. To build an accurate model with the help from the unlabeled songs and noisy tags, we present **SMART**, which stands for **S**emi-Supervised **M**usic **A**ffective Emotion **R**ecognition with Social **T**agging, combining of a graph-based semi-supervised learning algorithm with a novel tag refinement method. Experiments on the Million Song Dataset show that our proposed approach, trained with only 10 labeled instances, is as accurate as Support Vector Regression trained with 750 labeled songs.

1 Introduction.

Music emotion[1] recognition (MER) aims to extract emotion information from the musical content, which has attracted more and more attention in recent years, since it is essential for many important and interesting tasks, e.g. automatic soundtrack generation for online videos, images, etc. [1] , music recommendation [2] and emotion-based mobile device music players , e.g., Mood Pal[2]. Armed with the MER technique, users can retrieve and index music according to their moods.

In psychology, emotion is usually represented in a two-dimension-emotion plane [3]. Typically, MER is formulated as a classification problem by separating the emotion plane into a few categories [4, 5], or a regression problem which predicts the coordinate position of music in the emotion plane [6, 7]. Formulating MER as a regression problem is generally considered more appropriate because it resolves the ambiguity issue, as no categorial classes are needed [6].

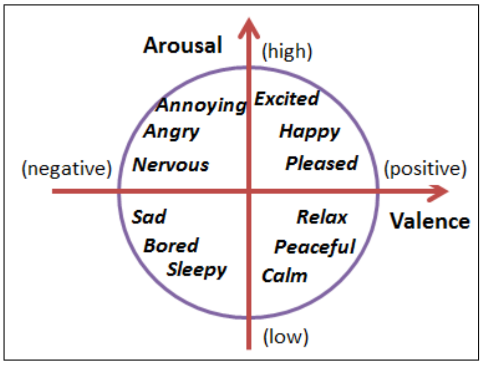


Figure 1: Thayer's valence-arousal emotion plane.

Although several approaches have been proposed to solve the MER problem by supervised learning methods [4, 6–8], none are genre-aware. They build accurate models relying on a large amount of labeled data. In practice, however, labeled data is very hard to obtain in most genres except for Pop music [9] (automatic indexing systems can still be built given a genre [10]). This is because, human labor and "expert time" iare expensive. Recently researchers have found that emotion is genre specific [11] in music, which means that we can neither apply the model trained on one genre to another nor build accurate models based on the limited labeled data for a particular genre, (e.g., it would be not possible to infer the emotion of a Classical song based on Pop songs). Therefore, the MER problem should be viewed from different perspective. Fortunately, with the development of Web 2.0, many music websites have been launched that provide a great deal of music-related data such as MP3, tags. These can shed light on how to solve the data sparsity problem in MER.

First, compared to the limited number of labeled songs with emotion, there are plenty of unlabeled songs on music websites which can be easily obtained, e.g., free music on Last.fm. Under a reasonable assumption that songs which have similar contents tend to have similar emotional labels [12], we can iteratively predict

*{bwuaa,ezhong,derekhh,horner,qyang}@cse.ust.hk, Department of Computer Science and Engineering, Hong Kong University of Science and Technology

[1] "Emotion" and "mood" are often used interchangeably. In this paper, we use "emotion".

[2] http://itunes.apple.com/us/app/mood-pal-daily-music-emotions/id492242448?mt=8

and assign labels by propagating them from labeled to unlabeled data that are similar, and thereby increase the amount of labeled data.

Second, on most music websites, such as Last.fm, users can create tags[3] for songs and these tags are emotion-related. The challenge in exploiting social tags is that tagging is socially-distributed and inherently noisy. Emotions tagged by casual users may be different from those by experts. For example, sadness in music is rarely perceived as an unpleasant emotion by casual users [13]. As shown in Figure.2, 'Yesterday Once More' was labeled as 'Sad' in the All Music Guide[4] (AMG), but only received a small number of 'Sad' tags in Last.fm. Moreover, in social tagging, there are some spammers or novices who simply cannot recognize a song emotion. Again, 'Yesterday Once More', has both 'Sad' and 'Happy' tags, which are totally opposite. Though the perceived emotion can depend on the listener's mood, we can also argue that tags created by casual users cannot be fully trusted. The tagging information needs to be further filtered and denoised before being used in a model. As these tags are uncertain and noisy by nature, in this paper we denote songs with emotion-related social tags as *noisy-label data*.

To address these issues, we present **SMART**, which stands for **S**emi-Supervised **M**usic **A**ffective Emotion **R**ecognition with Social **T**agging. It makes use of both unlabeled and noisy-label data. First, to utilize the knowledge of unlabeled data, we introduce a graph-based semi-supervised algorithm (GSSL) to propagate labels from labeled to unlabeled songs. The main idea is to assign labels transductively according to content similarities. Second, we propose a tag refinement method in order to adjust song-tag relations by learning a refinement matrix.

Our experimental results show that GSSL outperforms Support Vector Regression in Mean Square Error (MSE) by 10% for valence and 2.5% for arousal. In addition, the proposed method with 10 labeled instances can perform as well as Support Vector Regression with 750 labeled examples. Furthermore, after refining the song-tag correlation, **SMART** improves slightly (0.2% for valence and 0.5% for arousal). More importantly, in a noisy environment, where many tags' emotion values do not agree with their related songs, the improvement increases to 1.3% for valence and 1.2% for arousal, which indicates the robustness of **SMART**.

Table 1: Notation

Notation	Definition
l	Number of labeled songs
u	Number of unlabeled songs
n	Number of songs $n = l + u$
Y_L	Labeled songs' emotion value vector
Y_U	Unlabeled songs' emotion value vector
m	Number of social tags
T	Vector of social tags
$\boldsymbol{e}$	Social tags' emotion value vector
R	Correlation matrix between songs and tags
r_{ik}^T	Correlation between the i-th song and the k-th tag

2 Problem Definition.

We formulate the problem in this section. The notation list can be found in Table 1.

Briefly, MER works as follows. Given an input song, the MER system outputs the corresponding emotion. As shown in Figure 1, the valence (X) axis describes how positive (high valence) or negative (low valence) the emotion is, while the arousal (Y) axis describes how excited (high arousal) or calm (low arousal) the emotion is. Each word in the plane can be represented as a real number tuple (v,a) where v is the degree of valence and a is the degree of arousal. The range of the scalar is usually 0-10 [14].

Let $\{(x_1, y_1), \ldots, (x_l, y_l)\}^T$ be labeled songs, where $Y_L = \{y_1, \ldots, y_l\}^T \in \mathbb{R}^{l\times 2}$ is the label set indicating the emotion values of the songs. The emotion value is a tuple (v, a) denoting the emotion values of valence and arousal respectively.

Let $\{(x_{l+1},\ y_{l+1}), \ldots, (x_{l+u}, y_{l+u})\}^T$ be unlabeled songs, where $Y_U = \{y_{l+1}, \ldots, y_{l+u}\}^T$ denotes the unobserved labels of the unlabeled data, and $l \ll u$. Let $Y = Y_L \cup Y_U$, denote the complete set of songs. Let $X = \{x_1, \ldots, x_{l+u}\}^T \in \mathbb{R}^{(l+u)\times d}$, which contains $l + u$ feature vectors extracted from the $l + u$ songs.

In addition, for each song (x_i, y_i), we have auxiliary knowledge from a set of tags $T = \{(t_1, e_1), \ldots, (t_m, e_m)\}^T$ assigned by users on social media websites, where $\boldsymbol{e} = \{e_1, \ldots, e_m\}^T \in \mathbb{R}^{m\times 2}$ is a set of real numbers indicating the true emotion values of the tags $\{t_1, \ldots, t_m\}^T$. The correlation between (X, Y) and $\boldsymbol{e}$ can be represented as a correlation matrix $R \in \mathbb{R}^{m\times(l+u)}$, $i = 1, \ldots, l + u$, where r_{ki}, $1 \le k \le m$, represents the correlation between the k-th tag and the i-th song. Note that $\boldsymbol{e}$ and Y share the same value space. The emotion values are projected to Thayer's model [3]. Then, our task is to predict Y_U based on X and Y_L with additional knowledge in the form of T and R.

[3] In this paper, we refer to "tags" as words used by casual users online to describe music, while "labels" are words used by experts to describe music from a professional point of view.

[4] "The All Music Guide", Available: http://www.allmusic.com.

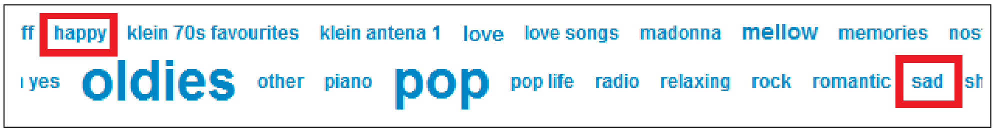


Figure 2: Screenshot of the tag list for 'Yesterday Once More' from Last.fm, where 'Happy' and 'Sad' were both tagged by users. This song is labeled as 'Sad' in AMG, but very few people selected this tag (only 1.6% according to MSD, while it received nearly the same number for 'Happy').

3 Semi-supervised Regression with Tag Refinement

Two challenges need to be addressed in the proposed model: 1) How to exploit knowledge in the unlabeled data? 2) How to refine song-tag relations in order to reduce the effect of uncertainty and noise in the tags?

For the first challenge, each song has some emotionally similar songs and tags (some tags could be noisy). The similarities between songs can be obtained by the similarities of their audio features. The weights of correlation between songs and tags can be obtained by the number of people who assign a certain tag to a certain song. With the similarity relations, we can construct a graph containing songs and tags, where those similar ones tend to connect with each other and thereby become neighbors. Intuitively, each song's emotion is similar to its neighbors' and therefore although the number of songs with labels is limited, we can still use the similarity correlations to propagate supervision knowledge from labeled to unlabeled songs. Formulating songs, tags, and the song-tag correlations as a graph, we exploit a graph-based semi-supervised learning method to predict the unlabeled songs' labels Y_U using audio features X and tags' emotion values $\boldsymbol{e}$.

For the second challenge, our basic idea is to construct a better song-tag correlation matrix R' based on the original R. We can assume that there exists a projection matrix $\mathbf{W}$ such that $R'^T = R^T\mathbf{W}$, and our objective is to learn the projection matrix $\mathbf{W}$ under the supervision of the labeled songs Y_L in order to strengthen similar tags and weaken opposites. Then we can make another prediction of Y_U using the refined song-tag correlation matrix R'.

After solving the two challenges, we combine the two predictions with a tradeoff parameter. Formally, we have:

$$(3.1) \qquad Y_U = \lambda f_{se}(\boldsymbol{x}_i, \boldsymbol{r}_{i.}^T) + (1-\lambda) f_{su}(\boldsymbol{r}_{i.}^T)$$

where $0 \leq \lambda \leq 1$ is a tradeoff parameter, $f_{se}(\boldsymbol{x}_i, \boldsymbol{r}_{i.}^T)$ and $f_{su}(\boldsymbol{r}_{i.}^T)$ represent predictions from semi-supervised learning and song-tag based supervised learning parts, respectively, where $\boldsymbol{r}_{i.}$ denotes the i-th row of R. Here we use a linear combination of supervised and semi-supervised learning. Although R' refines the song-tag relationship compared to the original matrix R, there are still cases where R' in fact deviates from the ground truth. In such a case, if we build the semi-supervised learning framework using the refined R', the overall prediction performance would degrade, which we found in our experiments. Therefore, to improve robustness, we use a linear combination so that it can not only improve the performance of the model most of the time but also reduce performance degradation.

We discuss these two parts in the following subsections in detail.

3.1 Graph-based Semi-supervised Learning In Eq. (3.1), $f_{se}(\boldsymbol{x}_i, \boldsymbol{r}_{i.}^T)$ generates predictions based on labeled and unlabeled songs, and noisy-label data. It predicts using labeled songs Y_L, the similarity matrix between songs S, the song-tag correlation matrix R, and the tags' emotion value vector $\boldsymbol{e}$. S is calculated by audio features of each song, which will be defined later. The formula for f_{se} is:

$$(3.2) \qquad f_{se}(\boldsymbol{x}_i, \boldsymbol{r}_{i.}^T) = \gamma S_{i.}Y + (1-\gamma)\boldsymbol{r}_{i.}^T\boldsymbol{e}$$

where $\boldsymbol{x}_i$ is the i-th song and $\boldsymbol{r}_{i.}^T$ is the i-th row vector of R^T. As stated previously, we model $f_{se}(\boldsymbol{x}_i, \boldsymbol{r}_{i.}^T)$ as a graph-based regression model [12]. In the graph, we have two types of vertices, namely songs and tags, which correspond to (X, Y) and T. Some are labeled and others are unlabeled, which correspond to Y_L and Y_U respectively. There are also two types of edges, song-song relations, and song-tag relations. The song-tag relations are related to the correlation matrix R, which will be discussed in detail in Section 4.1. We model the song-song relations as a similarity matrix S, which is defined as $S = [s_{ij}]_{n \times n} \in \mathbb{R}^{n \times n}, i, j = 1, \ldots, n, i \neq j$, where $s_{ij} = sim(\boldsymbol{x}_i, \boldsymbol{x}_j), s_{ii} = 0, i = 1, \ldots, n$. $sim()$ is a similarity function defined by the Mahalanobis distance between two songs' audio feature vectors [15]:

$$(3.3) \qquad sim(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp\{-(\boldsymbol{x}_i - \boldsymbol{x}_j)\Sigma^{-1}(\boldsymbol{x}_i - \boldsymbol{x}_j)^T\}$$

where Σ is the covariance matrix of features across all songs, approximated as a diagonal matrix where the diagonal values are the variances of individual features. Afterwards, each row of S is normalized to sum to 1. Then, we can obtain the objective function:

$$(3.4) \qquad E_{se} = \sum_{i=l+1}^{n} \| Y_{Ui} - (\gamma S_{i.}Y + (1-\gamma)\boldsymbol{r}_{i.}^T\boldsymbol{e}) \|^2$$

where E_{se} is the prediction error of the semi-supervised learning part. Here, we use gradient descent to optimize this objective function [16]. Taking the partial derivative of E_{se} with respect to Y_{Ui}, we have:

$$\frac{\partial E_i}{\partial Y_{Ui}} = Y_{Ui} - (\gamma S_{i.}Y + (1-\gamma) r_{i.}^T \boldsymbol{e}) \tag{3.5}$$

where E_i denotes the semi-supervised learning error of Y_{Ui}. Then, we have the update rule for Y_{Ui}:

$$\triangle Y_{U_i}^{(k)} = Y_{U_i}^{(k)} - (\gamma S_{i.} Y^{(k)} + (1-\gamma) r_{i.}^T \boldsymbol{e}) \tag{3.6}$$

$$Y_{U_i}^{(k+1)} \leftarrow Y_{U_i}^{(k)} - \rho \triangle Y_{U_i}^{(k)} \tag{3.7}$$

where ρ is the learning rate of gradient descent. The proof of convergence of gradient descent can be found in [16].

Initially, each vertex maintains a (v, a) tuple to denote their labels. Labeled songs and tags are defined according to Y_L and $\boldsymbol{e}$ respectively, while unlabeled songs are initialized as a zero vector. Then, we can iteratively update Y_U until convergence.

3.2 Closed Form Solution

In this subsection, we will give the closed form solution for GSSL. Since we update Y_U once each iteration, we combine and reshape Eqs. (3.7) and (3.6). Note that in calculating Y_U, we only need to consider entries corresponding to $\{(x_{l+1}, y_{l+1}), \ldots, (x_{l+u}, y_{l+u})\}^T$, therefore we use $S_U \in \mathbb{R}^{u \times n}$ instead of S:

$$\begin{aligned} Y_U^{(k+1)} =& (1-\rho)Y_U^{(k)} \\ & + \rho(\gamma S_U Y^{(k)} + (1-\gamma) R_U^T \boldsymbol{e}) \\ =& (1-\rho) Y_U^{(k)} \\ & + \rho(\gamma \begin{bmatrix} S_{UL} & S_{UU} \end{bmatrix} \times \begin{bmatrix} Y_L \\ Y_U \end{bmatrix}^{(k)} \\ & + (1-\gamma) R_U^T \boldsymbol{e})) \end{aligned} \tag{3.8}$$

where $S_{UL} \in \mathbb{R}^{u \times l}$ and $S_{UU} \in \mathbb{R}^{u \times u}$ denote the unlabeled-labeled songs and unlabeled-unlabeled songs' similarity matrices respectively.

Then, Eq. (3.8) can be rewritten as:

$$\begin{aligned} Y_U^{(k+1)} =& (1-\rho) Y_U^{(k)} \\ & + \rho\gamma(S_{UU} Y_U^{(k)} + S_{UL} Y_L^{(k)}) \\ & \rho(1-\gamma) R_U^T \boldsymbol{e}) \\ =& ((1-\rho)\mathbf{I}_u + \rho\gamma S_{UU}) Y_U^{(k)} \\ & + \rho(\gamma S_{UL} Y_L + (1-\gamma) R_U^T \boldsymbol{e}) \\ =& \mathbf{A} Y_U^{(k)} + \mathbf{B} \end{aligned} \tag{3.9}$$

where $\mathbf{A} = ((1-\rho)\mathbf{I}_u + \rho\gamma S_{UU})$, $\mathbf{B} = \rho(\gamma S_{UL} Y_L + (1-\gamma) R_U^T \boldsymbol{e})$, and $\mathbf{I}_u$ is the identity matrix. Solving the recursion, we have:

$$\begin{aligned} Y_U^{(k+1)} =& \mathbf{A}^{k+1} Y_U^{(0)} \\ & + (\mathbf{A}^k + \mathbf{A}^{k-1} + \cdots + \mathbf{A} + \mathbf{I}_u)\mathbf{B} \\ =& \mathbf{A}^{k+1} Y_U^{(0)} + \frac{\mathbf{A}^{k+1} - \mathbf{I}_u}{\mathbf{A} - \mathbf{I}_u} \mathbf{B}. \end{aligned} \tag{3.10}$$

Limiting k to infinity, we have:

$$\lim_{k \to \infty} Y_U^{(k)} = \lim_{k \to \infty} \mathbf{A}^k Y_U^{(0)} + \frac{\mathbf{A}^k - \mathbf{I}_u}{\mathbf{A} - \mathbf{I}_u} \mathbf{B} \tag{3.11}$$

which is only valid when $\mathbf{I}_u - \mathbf{A}$ is invertible. Note that $\mathbf{A} = (1-\rho)\mathbf{I}_u + \rho\gamma S_{UU}$, $\rho \leq 1, \gamma < 1$ and each element in similarity matrix S_{UU} is less than or equal to 1. Furthermore, since each row of S_{UU} is normalized, the sum of the i-th row of A is:

$$\sum_j^u \mathbf{A}_{ij} = 1 - \rho + \rho\gamma \sum_j^u (S_{UU})_{ij} \leq 1 - \rho + \rho\gamma < 1 \tag{3.12}$$

Therefore $\exists \eta < 1$, such that $\sum_j (\mathbf{A}^k)_{ij} \leq \eta^k$ for any row $i$in $\mathbf{A}^k$ [17] and each element $(\mathbf{A}^k)_{ij} \leq \sum_j (\mathbf{A}^k)_{ij} \leq \eta^k$ converges to zero, i.e., $\lim_{k \to \infty} \mathbf{A}^k = \mathbf{0}$. Thus, we obtain the closed form solution:

$$\lim_{k \to \infty} Y_U^{(k)} = \frac{\mathbf{I}_u}{\mathbf{I}_u - \mathbf{A}} \mathbf{B} \tag{3.13}$$

Since there exists uncertainty in R, we still need one more step to improve the usage of the noisy social tagging. In the next section, we address this problem by proposing a novel Tag Refinement (TR) approach.

3.3 Tag Refinement

In Eq.(3.1), $f_{su}(\boldsymbol{r}_{i.}^T)$ generates predictions based on song-tag relations and emotion values related to the tags. Since the social tagging data may be uncertain and noisy, and the distribution and preference of social tagging by casual users are different from experts, we need to refine the song-tag correlation matrix R. Our basic idea is to learn a better song-tag correlation matrix R' by refining the original correlation matrix R. Therefore, we propose to learn a projection matrix $\mathbf{W}$ in order to determine the correlation between R and $\boldsymbol{e}$, such that:

$$R'^T = R^T \mathbf{W} \tag{3.14}$$

$$f_{su}(\boldsymbol{r}_{i.}^T) = \boldsymbol{r}_{i.}^T \mathbf{W} \boldsymbol{e} \tag{3.15}$$

where $\mathbf{W} \in \mathbb{R}^{m \times m}$. Therefore, we can obtain the objective function which minimizes the square loss between the emotion labels Y_L and the refined predictions based on R and $\boldsymbol{e}$:

$$\min_{\mathbf{W}} \sum_{i=1}^{\ell} \frac{1}{2}(Y_{Li} - \boldsymbol{r}_{i.}^T \mathbf{W} \boldsymbol{e})^2 + \frac{\beta}{2}\|\mathbf{W}\|_F^2$$

(3.16)

$$= \min_{\mathbf{W}} \sum_{i=1}^{\ell} \frac{1}{2}(Y_{Li} - \mathbf{F}\mathbf{W})^2 + \frac{\beta}{2}\|\mathbf{W}\|_F^2$$

where $\mathbf{W} \in \mathbb{R}^{m\times m}$ is a projection matrix, $R_L \in \mathbb{R}^{m\times \ell}$ is the song-tag correlation matrix of the labeled songs, β is a regularization parameter, and $\mathbf{F} = \boldsymbol{e} \otimes R_L \in \mathbb{R}^{m^2\times \ell}$. Note that $\mathbf{W}$ can be obtained in closed form [18] :

$$\mathbf{W}' = (\beta \mathbf{I} + \mathbf{F}\mathbf{F}^T)^{-1}\mathbf{F}\boldsymbol{y} \tag{3.17}$$

From the above equation, we obtain $\mathbf{W}$ as a matrix of type $\mathbb{R}^{m^2\times 1}$, and we need to reshape $\mathbf{W}'$ by dividing the m^2 elements into m groups and put each group into a new column, which reshapes $\mathbf{W}'$ into $\mathbf{W} \in \mathbb{R}^{m\times m}$. However, with this approach, $\mathbf{F}$ may have very large dimensions even if the data set is not large, so we can approximate $\mathbf{W}$ by the product of low-rank matrices [19] $\mathbf{U}\mathbf{V}^T$, where $\mathbf{U} \in \mathbb{R}^{m\times p}$ and $\mathbf{V} \in \mathbb{R}^{m\times p}$,

(3.18)

$$\min_{\mathbf{U},\mathbf{V}} \sum_{i=1}^{l} \frac{1}{2}(y_i - \boldsymbol{r}_{i.}^T \mathbf{U}\mathbf{V}^T \boldsymbol{e})^2 + \alpha_u \|\mathbf{U}\|_F^2 + \alpha_v \|\mathbf{V}\|_F^2$$

$$= \frac{1}{2}\|Y_L - R^T \mathbf{U}\mathbf{V}^T \boldsymbol{e}\|_F^2 + \alpha_u \|\mathbf{U}\|_F^2 + \alpha_v \|\mathbf{V}\|_F^2$$

where p can be a relatively small number and α_u, α_v are regularization parameters. Similar to $\mathbf{W}$, we can obtain the update equations for $\mathbf{U}$ and $\mathbf{V}$:

$$\mathbf{F}_{\mathbf{U}} = \mathbf{V}^T \boldsymbol{e} \otimes R_L \tag{3.19}$$

$$\mathbf{U} = (\alpha_u \mathbf{I} + \mathbf{F}_{\mathbf{U}}\mathbf{F}_{\mathbf{U}}^T)^{-1}\mathbf{F}_{\mathbf{U}} Y_L \tag{3.20}$$

$$\mathbf{F}_{\mathbf{V}} = \boldsymbol{e} \otimes \mathbf{U}^T R_L \tag{3.21}$$

$$\mathbf{V} = (\alpha_v \mathbf{I} + \mathbf{F}_{\mathbf{V}}\mathbf{F}_{\mathbf{V}}^T)^{-1}\mathbf{F}_{\mathbf{V}} Y_L \tag{3.22}$$

where $\mathbf{F}_{\mathbf{U}} \in \mathbb{R}^{mp\times l}$ and $\mathbf{F}_{\mathbf{V}} \in \mathbb{R}^{mp\times l}$. Note that both $\mathbf{V}$ and $\mathbf{U}$ should be reshaped like $\mathbf{W}$ after each update. Once we learned $\mathbf{U}$ and $\mathbf{V}$, we can make prediction:

$$f_{su}(\boldsymbol{r}_{i.}^T) = \boldsymbol{r}_{i.}^T \mathbf{W} \boldsymbol{e} \tag{3.23}$$

4 Experiments

We empirically answer the following questions in this section. 1) Does the proposed graph-based semi-supervised approach GSSL achieve similar or even less music emotion recognition error with fewer training samples by utilizing the unlabeled and noisy-label data? 2) Does the tag refinement solution change the song-tag relations? 3) How do the model parameters (e.g., number of neighbors in GSSL) influence the model performance? To answer these questions, we first introduce the datasets we used. We then describe what and how audio features were extracted. We then compare the performance of our method to some state-of-the-art methods. We will use Mean Square Error (MSE) as the evaluation metric for 1) and 2), which will be described in details in Section 4.3. We also conducted a parameter study was conducted to reflect the intrinsic properties of our methods.

4.1 Datasets Since the data is currently sparse in other genres, we cannot use them for evaluation. As labeled Pop data is sufficient, if the proposed method can train an accurate model with a small number of Pop training examples, the model could also work in other genres. Therefore, we make use of large amount of data in Pop music for evaluation purposes. We construct the evaluation dataset based on several real-world datasets.

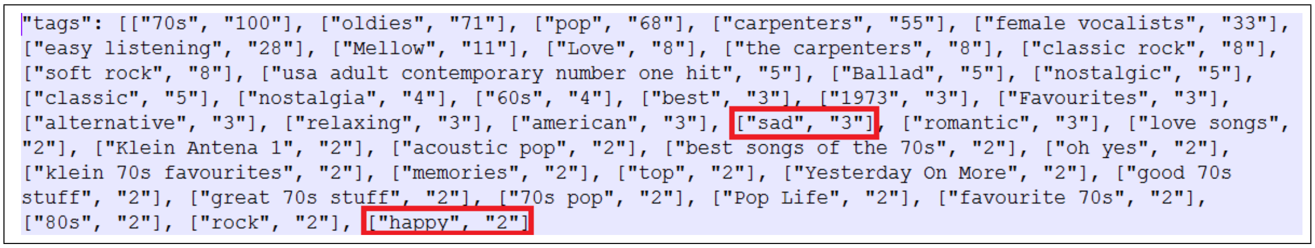
Affective Norms for English Words (ANEW) is a ‘word-emotion value’ table which consists of over 1000 words scored by subjects on how they feel about the words in terms of valence and arousal using real numbers from 0 to 10. We use ANEW as the emotional word list.

The All Music Guide (AMG) lists thousands of Pop songs labeled by experts with emotional words. We used these songs and labels as the ground truth. There are a total of 183 emotional labels and 5106 “label-song” entries. Note that some songs have multiple labels. In this paper, since the labels of one specific song are relatively close to each other (0.28 and 0.42 in standard deviation for valence and arousal respectively.), we simply assign the average value as the label to the song. After the aggregation, we have a total of 2914 songs.

The Last.fm Dataset[5] has over 500,000 Pop songs each with at least one social tag and over 500 thousand unique tags. We use this dataset as the source for social tags. Matching songs with the AMG and the Last.fm Dataset according to artist name and song title, we were left with 836 songs.

Specifically, the song-tag correlation matrix is constructed as follows. The Last.fm Dataset provides each song’s social tag list and the tags’ corresponding degrees of importance which is decided by how many people tagged a tag to the song. The tag list and corresponding degree of importance for ‘Yesterday once more’ is listed in Figure 3. We first filter these tags using ANEW, then normalize the weight of each tag such that the sum of the remaining tags’ weights equals 1. After the pre-processing step, we can get the song-tag correlation matrix R.

[5] http://labrosa.ee.columbia.edu/millionsong/lastfm



```
"tags": [["70s", "100"], ["oldies", "71"], ["pop", "68"], ["carpenters", "55"], ["female vocalists", "33"],
["easy listening", "28"], ["Mellow", "11"], ["Love", "8"], ["the carpenters", "8"], ["classic rock", "8"],
["soft rock", "8"], ["usa adult contemporary number one hit", "5"], ["Ballad", "5"], ["nostalgic", "5"],
["classic", "5"], ["nostalgia", "4"], ["60s", "4"], ["best", "3"], ["1973", "3"], ["Favourites", "3"],
["alternative", "3"], ["relaxing", "3"], ["american", "3"], ["sad", "3"], ["romantic", "3"], ["love songs",
"2"], ["Klein Antena 1", "2"], ["acoustic pop", "2"], ["best songs of the 70s", "2"], ["oh yes", "2"],
["klein 70s favourites", "2"], ["memories", "2"], ["top", "2"], ["Yesterday On More", "2"], ["good 70s
stuff", "2"], ["great 70s stuff", "2"], ["70s pop", "2"], ["Pop Life", "2"], ["favourite 70s", "2"],
["80s", "2"], ["rock", "2"], ["happy", "2"]
```

Figure 3: The tag list and corresponding degree of importance for 'Yesterday Once More' extracted from the Last.fm Dataset. The importances of 'happy' and 'sad' is 2 and 3 respectively (the two words wrapped with red rectangles have similar degree of importance.)

Finally, the remaining emotional labels and tags were in the following ranges: valence $\in [1.25, 8.82]$, arousal $\in [2.39, 8.17]$. We were left with 836 songs and 376 emotional social tags from Last.fm with an average of 4.1 tags for each song. We also connected the 836 songs with the Million Song Dataset (MSD) [20] by matching track IDs to extract audio features.

4.2 Musical Features To describe the audio features from a music emotion perpective, we consider several types of audio features, including timbre, pitch, rhythm, and loudness, which have been found to be effective features for describing music emotion [9].

MSD provides Mel-Frequency Cepstral Coefficient (MFCC) like features with 12 bands, including timbre and pitch. Also, we use the maximum loudness of each segment to outline the overall loudness of the music. For these three features, we take the mean and Standard Deviation (STD) every 5 seconds (i.e. the texture window), and take the mean and STD again for the complete song [21]. For rhythm, we use the overall tempo (beats per minute) and take the STD of beat intervals to represent the tempo regularity [7].

Finally, aggregating all four types of features, we have a 102 dimension feature vector in total.

4.3 Performance of GSSL The emotion predictions of the baseline and the proposed methods were evaluated by Mean Squared Error (MSE). We first compared the Graph-based Semi-supervised Learning (GSSL) model to the state-of-the-art approach, Support Vector Regression (SVR) [6, 15].

The musical audio features of SVR are the same as GSSL, and are denoted as 'Audio' in our results when we train the model simply based on the audio features. We exploit social tagging data in two representations. In the first, we use a weighted sum of tags' emotion values for each song, namely $\boldsymbol{r}_{i.}^T\boldsymbol{e}$ for the i-th song. We denote it as 'TEV' (tag emotion values). The second representation is bag-of-words (BOW), with term frequency-inverse document frequency weighting (tf*idf) [22] between tags and songs, namely $\boldsymbol{r}_{i.}^T$ for the i-th song. We denote it as 'BOW'. We provide another BOW tf*idf representation by clustering tags into 11 classes according to the division method of Thayer's emotion plane [7]. We denote it as 'BOW-c' (BOW-clustered).

To verify our hypothesis that we can make use of a limited amount of labeled data to predict a large amount of unlabeled data, we use a reverse ten-fold crossvalidation evaluation framework in the training process. We use one fold for training and the other nine for testing. Table 2 lists the experimental results. We also list 'Audio' and 'TEV' for comparison. In SVR, we use libsvm [23] , ϵ-SVR and a Radial Basis Function kernel.

The results in Table 2 show that for SVR, 'TEV' yields the best performance for valence while 'Audio+TEV+BOW' achieves the best prediction for arousal. For GSSL, 'Audio+TEV+BOW', which means adapting tags as vertices and song-tag relations as edges, yields the best performance. The results also reflect the general observation that valence recognition is much more challenging than arousal [9]. More importantly, GSSL outperforms SVR (with 'TEV') by 10% (0.344 in MSE) in valence and 2.2% (0.04 in MSE) in arousal.

We also examined the performance of both methods with respect to different numbers of training instances. In Figure 4, we compare GSSL and SVR performance when trained with 1%, 10%, 50% and 90% of the instances. From the valence result, the performance of GSSL is surprisingly good. Trained with only 1% (about 10 instances), it already performs as well as SVR trained with 90% (about 750 instances). The reason may be that SVR cannot make use of the informative unlabeled data. Furthermore, in SVR, tags are treated as part of the features, while in GSSL they can be treated as noisy-labels which can be propagated within the graph. For arousal, although the performance of GSSL is not as surprising as for valence, it still beats SVR with only

Table 2: Comparison between SVR and GSSL trained with 10% instances using MSE. Values in bold are the best performance within valence/arousal. SVR(V/A): MSE of SVR for valence/arousal; GSSL(V/A): MSE of GSSL for valence/arousal.

	SVR (V)	GSSL (V)	SVR (A)	GSSL (A)
Audio	4.71622	3.76743	1.82305	1.83613
TEV	3.78669	3.60061	1.86007	2.32252
BOW	5.35682	N/A	1.98210	N/A
BOW_ clustered	5.04255	N/A	1.87656	N/A
Audio+TEV+BOW	4.70258	**3.44207**	1.81411	**1.77333**
Audio+TEV+BOW-c	4.71522	N/A	1.82249	N/A

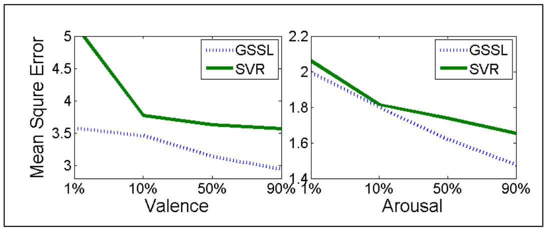


Figure 4: MSE of both methods with respect to different number of training instances. The total number of instances is 836.

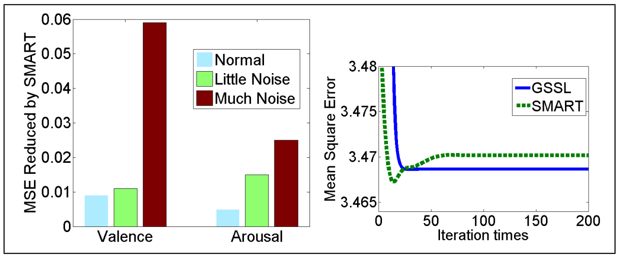


Figure 5: Reduction in MSE under different modes (left, higher is better) and MSE with respect to iteration times (right)

50% training instances (about 420 instances) while SVR used 90% of the labeled instances.

In summary, the above results reflects two facts: 1) Exploiting unlabeled data is valuable and effective, 2) social tagging helps improve the prediction because it can reflect the emotion of the songs. Social tagging information can be useful features in SVR and can be treated as 'noisy-labels' in GSSL. However, can the tags be used directly as labels? We'll discuss this by examining the effectiveness of our proposed **SMART** approach in the next subsection.

4.4 Performance of SMART Since social tagging may be noisy and uncertain, we investigated the performance of **SMART**. We make use of R as the correlation matrix and we denote this method as 'Normal'. We added noises in the correlation matrix to verify the performance under increasingly noisy environments. We manually constructed two noisy datasets, which are denoted as 'Little Noise' and 'Much Noise' respectively. In 'Little Noise', we deleted the best 10% of tags for each song. In 'Much Noise', we selected the worst tags assigned by users and raised the weight (to around 0.4 after normalization). Table 3 shows the results.

From Table 3, we found that **SMART** works better than GSSL. This is because **SMART** refines the song-tag relations and improves the confidence. In 'Normal', 'Little Noise' and 'Much Noise', **SMART** reduces the MSE by (0.009, 0.005), (0.011, 0.015), and (0.059, 0.025) for valence and arousal respectively, which is shown more clearly in Figure 5, where the Y-axis is the MSE reduced by **SMART**. We can come to two conclusions: 1) There exists noise in social tagging. 2) Our method is able to filter and denoise social tagging. We studied the tag refinement effect of song 'Pictures Of You' using the song 'The Cure', which is labeled as 'Sad' in AMG, under 'Normal' conditions. In Last.fm, the original weight for 'Sad' was 0.0612. After refinement, it was raised to 0.0786. The reason might be this song has neighbors which are tagged or labeled with similar emotion words such as 'Sad', 'Lost', bringing it closer to the ground truth.

4.5 Parameter Study Next, we studied the effectiveness of the various parameters in our model. First, we investigated the number of nearest-neighbors chosen to pre-process the similarity matrix S in GSSL. We incremented k from 1 to 20 and obtained the MSE results shown in Figure 6. In valence, the error is minimized when the number of nearest-neighbors decreases to 5, while for arousal, it achieves best performance when the number of nearest-neighbors is 10. We obtained the best performance when the number of nearest-neighbors was relatively small, which is a good sign for real-world applications since this will greatly decrease the number of edges in the learning graph and hence improve the efficiency of the algorithm.

Another important parameter in our approach is the number of iterations. We've compared the MSE

Table 3: Comparison of **SMART** and GSSL. The low rank dimension p is set to 2.

	GSSL(V)	**SMART**(V)	GSSL(A)	**SMART**(A)
Normal	3.44207	**3.43435**	1.77333	**1.76872**
Little Noise	3.65741	**3.64654**	1.86382	**1.84860**
Much Noise	4.75760	**4.69848**	1.96304	**1.93830**

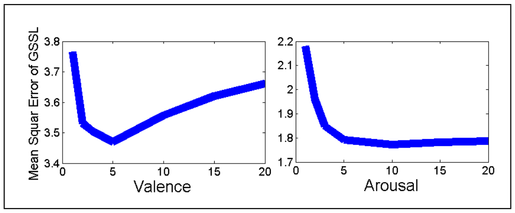


Figure 6: Mean Square Error with respect to the number of nearest-neighbors for valence and arousal

for various iterations between **SMART** and GSSL without Tag Refinement. As shown in Figure.5 we can see that both GSSL and **SMART** converge within about 25 iterations. The MSE of **SMART** rises after convergence because it jumps out of the convergence interval.

5 Related Work

Much works has been done on MER. Some supervised classification and regression methods have been applied to MER [1, 4, 6, 7]. [24] also investigated multi-label classification in MER. However, as music emotion is genre-specific, the data sparsity problem becomes serious. As far as we know, state-of-the-art methods rely on the amount of labeled data, which is a problem for real-world applications due to the scarcity of labeled data in most genres. Some researchers have also focused on extracting social tagging knowledge to help MER [25, 26]. However, according to Laurier *et al.*'s finding [27], inconsistencies exist between experts and casual users when using emotional words to describe music. Previous work has only considered tags as features and have not discussed their uncertain and noisy nature. Also, since the distribution of social tagging is different with emotional labels, we cannot simply train a model on social tagging data.

Graph-based Semi-supervised Learning has been widely studied. Stikic *et al.* designed a Multi-graph Based semi-supervised learning method for activity recognition. Goldberg *et al.* [28] used graph-based semi-supervised learning for sentiment categorization. Niu *et al.* [29] proposed to solve word sense disambiguation problem using Graph-based Semi-supervised Learning. In Music Information Retrieval, Li *et al.* [30] proposed to use semi-supervised learning to identify similar artists. However, to the best of our knowledge, semi-supervised learning has not yet been proposed to solve MER.

Tag refinement has also been studied on many applications. Liu *et al.* [31] studied on providing better tags for online images. Sang *et al.* [32] exploited user information to correct tags for images. However, their work sought to automatically assign better tags for images while ours aims to improve MER by refining the song-tag correlation and therefore different in purpose. Their methods are not directly applicable to solve the noisy-label problem in MER.

6 Conclusions and Future Works

Since MER is genre-specific, more research is required to exploit auxiliary knowledge and design methods to use such knowledge effectively. In this paper, we have exploited unlabeled and noisy-label data (social tagging) to help Music Emotion Recognition. Our work has two major contributions:

1) We have proposed a graph-based semi-supervised learning method to predict music emotion by making use of both unlabeled audio and social tagging information. Evaluation on Pop songs shows that the use of unlabeled and noisy-label data gives good results in genre-specific MER. Our method can make accurate predictions even if the number of training instances is small, i.e. 10, which beats the state-of-the-art method trained with 750 instances [6, 7]. This result is encouraging for MER tasks in other genres which lack labeled data (e.g., Classical and Jazz).

2) Since social tagging information plays a more important role when labeled data is scarce, we considered social tagging as noisy-label data and found that denoising and filtering rendered a better performance compared to naively treating social tagging as trustworthy knowledge. Moreover, the proposed method can withstand the onslaught of increasing level of noise well.

In the future, since semi-supervised learning may propagate errors in the learning process, we will consider to introduce active learning to help select instances that are valuable for labeling and thereby reduce manual labeling. Moreover, we may take user profiles and users' social networks into consideration to model the tagging abilities of users more accurately.

7 Acknowledgements

This work was supported by Hong Kong Research Grants Council grants 613510, 613111, and 613112. The authors would like to thank Dr. Nathan Liu and Dr. Weike Pan for their kind helps and suggestions. The authors also appreciate the anonymous reviewers for their reviewing and useful comments.

References

[1] A. Stupar and S. Michel, "Picasso: automated soundtrack suggestion for multi-modal data." in *CIKM*, 2011, pp. 2589–2592.

[2] R. Cai, C. W. L. Z. Zhang, Chao, and W.-Y. Ma, "Musicsense: contextual music recommendation using emotional allocation modeling." in *ACM MM*, 2007, pp. 553–556.

[3] R. Thayer, *The biopsychology of mood and arousal.* Oxford University Press, USA, 1989.

[4] L. Lu, D. Liu, and H. Zhang, "Automatic mood detection and tracking of music audio signals." *IEEE TASLP*, vol. 14, no. 1, pp. 5–18, 2006.

[5] J.-C. Wang, H.-S. Lee, H.-M. Wang, and S.-K. Jeng, "Learning the similarity of audio music in bag-of-frames representation from tagged music data." in *ISMIR*, 2011, pp. 85–90.

[6] Y.-H. Yang, Y.-C. Lin, Y.-F. Su, and H. H. Chen, "A regression approach to music emotion recognition." *IEEE TASLP*, vol. 16, no. 2, pp. 448–457, 2008.

[7] B. Jun Han, S. Rho, R. B. Dannenberg, and E. Hwang, "Smers: Music emotion recognition using support vector regression." in *ISMIR*, 2009, pp. 651–656.

[8] K. Bischoff, C. S. Firan, R. Paiu, W. Nejdl, C. Laurier, and M. Sordo, "Music mood and theme classification - a hybrid approach." in *ISMIR*, 2009, pp. 657–662.

[9] Y.-H. Yang and H. H. Chen, "Machine recognition of music emotion: A review," *ACM TIST*, vol. 3, pp. 1–10, 2012.

[10] J. Grekow and Z. W. Ras, "Emotion based midi files retrieval system." in *Advances in Music Information Retrieval*, ser. Studies in Computational Intelligence, Z. W. Ras and A. Wieczorkowska, Eds. Springer, 2010, vol. 274, pp. 261–284.

[11] T. Eerola, "Are the emotions expressed in music genre-specific? an audio-based evaluation of datasets spanning classical, film, pop and mixed genres," *Journal of New Music Research*, vol. 40, no. March 2012, pp. 37–41, 2011.

[12] O. Chapelle, B. Scholkopf, and A. Zien, *Semi-Supervised Learning.* MIT Press, 2006.

[13] T. Eerola, O. Lartillot, and P. Toiviainen, "Prediction of multidimensional emotional ratings in music from audio using multivariate regression models." in *ISMIR*, 2009, pp. 621–626.

[14] M. M. Bradley and P. J. Lang, "Affective norms for english words (anew): Instruction manual and affective ratings," *Psychology*, no. C-1, pp. 1–45, 1999.

[15] M. Mandel and D. Ellis, "Song-level features and support vector machines for music classification," in *ISMIR 2005*, 2005, pp. 594–599.

[16] J. Nocedal and S. Wright, *Numerical Optimization*, ser. Springer Series in Operations Research. Springer, 1999.

[17] X. Zhu, "Semi-supervised learning with graphs," Ph.D. dissertation, Carnegie Mellon University, 2005.

[18] W. Pan, N. N. Liu, E. W. Xiang, and Q. Yang, "Transfer learning to predict missing ratings via heterogeneous user feedbacks." in *IJCAI*, 2011, pp. 2318–2323.

[19] G. Golub and C. Van Loan, *Matrix computations.* Johns Hopkins Univ Pr, 1996, vol. 3.

[20] T. Bertin-Mahieux, D. P. W. Ellis, B. Whitman, and P. Lamere, "The million song dataset." in *ISMIR*, 2011, pp. 591–596.

[21] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE TSALP*, vol. 10, no. 5, pp. 293–302, 2002.

[22] X. Wang, X. Chen, D. Yang, and Y. Wu, "Music emotion classification of chinese songs based on lyrics using tf*idf and rhyme." in *ISMIR*, 2011, pp. 765–770.

[23] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM TIST*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[24] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. P. Vlahavas, "Multi-label classification of music into emotions," in *ISMIR*, 2008, pp. 325–330.

[25] Y.-C. Lin, Y.-H. Yang, and H. H. Chen, "Exploiting online music tags for music emotion classification." *TOMCCAP*, vol. 7, no. Supplement, p. 26, 2011.

[26] X. Hu and B. Yu, "Exploring the relationship between mood and creativity in rock lyrics." in *ISMIR*, 2011, pp. 789–794.

[27] C. Laurier, M. Sordo, J. Serr, and P. Herrera, "Music mood representations from social tags." in *ISMIR*, 2009, pp. 381–386.

[28] A. B. Goldberg, X. Zhu, and X. Zhu, "Seeing stars when there aren't many stars: Graph-based semi-supervised learning for sentiment categorization," in *HLT-NAACL 2006 Workshop on Textgraphs: Graph-based Algorithms for Natural Language Processing*, 2006.

[29] Z.-Y. Niu, D.-H. Ji, and C. L. Tan, "Word sense disambiguation using label propagation based semi-supervised learning." in *ACL*, 2005.

[30] T. Li and M. Ogihara, "Music artist style identification by semi-supervised learning from both lyrics and content." in *ACM MM*, 2004, pp. 364–367.

[31] Y. Liu, F. Wu, Y. Zhang, J. Shao, and Y. Zhuang, "Tag clustering and refinement on semantic unity graph." in *ICDM*, 2011, pp. 417–426.

[32] J. Sang, C. Xu, and J. Liu, "User-aware image tag refinement via ternary semantic analysis." *IEEE TMM*, vol. 14, no. 3-2, pp. 883–895, 2012.