Leveraging Affective Hashtags for Ranking Music Recommendations

Eva Zangerle, Chih-Ming Chen, Ming-Feng Tsai, and Yi-Hsuan Yang, *Senior Member, IEEE*

Abstract—Mood and emotion play an important role when it comes to choosing musical tracks to listen to. In the field of music information retrieval and recommendation, emotion is considered contextual information that is hard to capture, albeit highly influential. In this study, we analyze the connection between users' emotional states and their musical choices. Particularly, we perform a large-scale study based on two data sets containing 560,000 and 90,000 #nowplaying tweets, respectively. We extract affective contextual information from hashtags contained in these tweets by applying an unsupervised sentiment dictionary approach. Subsequently, we utilize a state-of-the-art network embedding method to learn latent feature representations of users, tracks and hashtags. Based on both the affective information and the latent features, a set of eight ranking methods is proposed. We find that relying on a ranking approach that incorporates the latent representations of users and tracks allows for capturing a user's general musical preferences well (regardless of used hashtags or affective information). However, for capturing context-specific preferences (a more complex and personal ranking task), we find that ranking strategies that rely on affective information and that leverage hashtags as context information outperform the other ranking strategies.

Index Terms—Emotion in music, emotion regulation, sentiment detection, ranking, music recommendation, microblogging, hashtags

1 Introduction

People listen to music for different reasons: to relieve from boredom, fill uncomfortable silences, social cohesion and communication, emotion regulation, etc. [1], [2]. From an affective computing point of view, it is interesting to investigate the relationship between a user's musical preference and the user's emotional state. There have been many psychological studies on the role of music in emotion regulation [2], [3], [4]. The emotional state of a listener has also been considered as important contextual information in building recommender systems [5], [6], [7]. A possible application is to build a system that monitors people's emotion and predicts how to subliminally impact them by recommending different music pieces. However, as the emotional state of a user is hard to capture in a large-scale study, most existing studies are conducted in a laboratory setting. It remains unclear to which extent such findings can be generalized to the real-life usage of music [8].

Seeing the popularity of social microblogging websites such as Twitter,[1] we have new opportunities to study real-world music listening behavior at scale [9], [10], [11], [12]. Most interestingly for our study, Twitter allows for gathering so-called #nowplaying tweets [9], which are tweets describing the track a user is currently listening to. One such example tweet is "#nowplaying Crazy For You by Adele #Happy". In this example, the user not only publishes the music track and artist he/she is listening to, but also adds a hashtag (i.e., keywords or phrases starting with the symbol #) describing his/her concurrent emotional state. Users add these hashtags spontaneously in real life, and there is an abundant number of such #nowplaying tweets with affect-related hashtags. We are therefore particularly interested in how the affective hashtags within a tweet are related to the user's musical preferences. For this purpose, we consider only #nowplaying tweets containing hashtags that represent some notion of emotion (i.e., contextual information), and aim to study their role in providing contextual affection-aware music recommendations tailored to the user's current emotional state and musical preferences. We have the following two research questions (RQs) to be answered:

- **RQ1:** How can affective contextual information contribute to improving personalized ranking of track recommendation candidates?
- **RQ2:** How can we computationally represent the affective contextual information in a #nowplaying tweet?

There has been excellent work on context-aware recommendation and representation learning [13], [14], [15], [16], [17], sentiment analysis from text [18], [19], [20], [21], [22], as well as emotion-based music recommendation [23], [24], [25], [26]. The main novelty of this study lies in the way we study the aforementioned two RQs by adapting existing techniques. Specifically, our study differentiates itself from the prior arts in the following aspects:

1. http://twitter.com

- *E. Zangerle is with the Department of Computer Science, University of Innsbruck, Innsbruck 6020, Austria. E-mail: eva.zangerle@uibk.ac.at.*
- *C.-M. Chen and M.-F. Tsai are with the National Chengchi University, Taipei 11605, Taiwan. E-mail: changecandy@gmail.com, mftsai@nccu.edu.tw.*
- *Y.-H. Yang is with Academia Sinica, Taipei 11529, Taiwan. E-mail: yang@citi.sinica.edu.tw.*

Manuscript received 20 Dec. 2016; revised 14 Feb. 2018; accepted 23 May 2018. Date of publication 12 June 2018; date of current version 1 Mar. 2021. (Corresponding author: Eva Zangerle.)
Recommended for acceptance by G. Ngai.
Digital Object Identifier no. 10.1109/TAFFC.2018.2846596

First, we propose to employ, and compare the result of two evaluation tasks to highlight the importance of contextual information (Section 3). For a given user and a context, the first task requires ranking the relevance of a set of tracks that are picked at random, whereas the second task requires ranking a set of tracks that are known (from the training set) to be associated with the user. While the first task is mainly about the *general preference* of a user (i.e., which tracks a user would like), the second task requires modeling the *context-specific preference* of a user for we already know that all the candidate tracks are liked by the user but only one of them can be ranked at top given that specific listening context. An algorithm cannot perform well if it does not know how a user's emotional context affects his or her musical preference. In comparison, existing work on context-aware recommendation usually focuses on the algorithms and simply takes the full catalog of data in their evaluation [14], [15], [16], [17]. Such an evaluation method does not distinguish between tracks that have been known to or not by users, making it hard to assess whether an algorithm learns the general preference or context-specific preference. This is less a concern for a general recommendation algorithm but is critical in addressing our RQs.

Second, to investigate the affective contextual information embedded in the #nowplaying tweets, we propose to treat the user-track-hashtag association as a graph and use state-of-the-art *network embedding* methods [27], [28], [29] to learn latent feature representations of users, tracks and hashtags (Section 4.2). By experimenting with different combinations of the representations (Section 4.3), we can test different assumptions about the underlying association between users and tracks. For example, a user can be represented by the user's own latent representation (denoted as "user"), but can also be represented by the average latent representation of the hashtags the user has used before in his or her tweets ("usertag"). Similarly, a track can be represented by its own latent representation ("track") or by the average representation of the hashtags the track has been associated with by different users ("tracktag"). As the hashtags are restricted to be affect-related ones, "usertag" and "tracktag" may respectively capture the *general emotional tendency* of a user and a track. A possible consequence is that, if a track is typically listened to in a specific emotional context across users, "tracktag" may outperform "track" in the above-mentioned second task, for "tracktag" encodes affective information in a more explicit way. In total, six ranking methods are considered. To our best knowledge, testing the representations in such an emotion-centered way has not been attempted before.

Third, in addition to the latent representations, we employ different sentiment dictionaries proposed in the literature of sentiment analysis [30], [31], [32], [33], [34], [35] to implement two ranking methods that solely rely on the sentiment scores (Section 4.1). In this way, we can study RQ2 using two approaches: based on the latent representations and based on the sentiment scores. Our experiments (Section 5) show that for the first task (capturing a user's general preferences), utilizing latent representations for users, tracks and hashtags contributes to better and more personalized ranking results. However, for the second, more complex and personal context-specific task, the sentiment-aware ranking methods outperform the other ranking methods. This finding implies that the more personal and complex a ranking task gets, the higher the influence and significance of affective information gets.

Finally, although emotion-based music recommendation is not new, existing work mostly relies on user data collected in a controlled environment and the scale is usually small [23], [24], [25]. In contrast, our study is based on a large collection of Twitter data (around 560 K) that contain real-world music listening information (Section 2). We will share the data with the research community for reproducibility and for promoting research in this direction.

TABLE 1
Data Set Statistics

Characteristics	Original [9]	#NP560k	#NP90k
Listening events	21,501,261	564,301	85,528
Tracks distinct	654,012	51,045	31,454
Artists distinct	79,011	8,210	8,020
Users distinct	176,909	9,431	9,336

2 Data Sets

Generally, we require a data set that provides information about the listening behavior and emotional states of users for conducting the proposed experiments. Therefore, we employ the #nowplaying data set compiled by Zangerle et al. [9] for the study, as this data set provides the required information. The data set is composed of #nowplaying tweets crawled via the Twitter API [36] and provides the timestamp when the tweet was sent, an anonymized user id, the tweet's source (how it was sent), the contained artist name and track title. An example listening event is: `< 2016-05-12 16:26:42, '7bd5237385a73c54265cd02aa136dbecdb88a0b8', 'Twitter Web Client', 'Hello, Goodbye', 'The Beatles'>`.

To gather a data set that allows for representative user profiles, we chose to extract all listening events of users who have sent a minimum of ten listening events in the years 2014 and 2015 from the #nowplaying data set. The characteristics of the resulting data set are shown in Table 1 (column "Original"). For our study, we focus on tweets for which we can detect a sentiment value by using the methods described in Section 4.1, as only this data allows to evaluate the influence of affective contextual information on the quality of track recommendation rankings. Therefore, we remove the listening events that do not contain any hashtag that we can obtain a sentiment score for, leading to a subset containing 564,301 listening events. Statistics of this *#NP560k* data set are listed in Table 1.

Table 2 presents the five-number-summaries describing the tagging and listening behavior of users within the #NP560k data set. We also list the tracks and listening events per user (overall and distinct) as well as the number of tags per user and per track (overall and distinct). We observe that while the maximum number of listening events per user, track and hashtag are very high, the mean, median and the 1st and 3rd quartile of these characteristics are substantially lower implying that these distributions are skewed and do not follow a normal distribution. Also, we observe a small number of users and tracks that feature profoundly higher numbers for the analyzed characteristics in

TABLE 2
Five-Number-Summaries of (Left) the #NP560k Data Set and (Right) the #NP90k Data Set

	#NP560k					#NP90k				
Characteristics	Median	Q3	Max	Mean	SD	Median	Q3	Max	Mean	SD
Listening events per user	2.0	4.0	69,197.0	59.83	1,125.48	2.0	4.0	463.0	9.16	31.70
Listening events per track	2.0	8.0	1,821.0	11.05	33.65	1.0	2.0	1,031.0	2.72	10.42
Tracks per user	2.0	4.0	69,197.0	59.83	1,125.48	2.0	4.0	463.0	9.16	31.70
Distinct tracks per user	2.0	4.0	3,500.0	10.95	78.73	2.0	4.0	319.0	6.26	19.03
Hashtags per user	2.0	5.0	86,855.0	74.10	1,446.10	2.0	5.0	1,025.0	10.84	39.27
Distinct hashtags per user	1.0	3.0	207.0	2.65	5.41	1.0	3.0	108.0	2.57	4.43
Hashtags per track	1.0	1.0	6.0	1.24	0.47	1.0	1.0	6.0	1.17	0.44
Distinct hashtags per track	1.0	1.0	6.0	1.23	0.46	1.0	1.0	6.0	1.16	0.44

We show (from left to right) the median, 3rd quantile (Q3), maximum, mean, standard deviation (SD) of the individual characteristics for both data sets. The minimum and 1st quantile for both data sets are all ones for all the characteristics.

comparison to the majority of users and tracks. Such heavy-tailed distributions have been shown to be prevalent in social networks [37], [38].

Due to the heavy-tailed characteristics of the #NP560k data set, we chose to introduce a second data set to investigate the role and impact of outlier users (i.e., users who feature profoundly larger number of listening events). To do this, we create a data set that is cleaned from those outlier users and is less skewed in terms of the number of listening per users than the #NP560k data set. Hence, we apply an outlier removal method to the #NP560k data set. Particularly, we keep all users within the 99th percentile of the distribution and remove the others, as this outlier removal method has been shown to be suited for highly skewed distributions [39]. This presents us with a smaller data set, referred to as the *#NP90k* data set in this paper. Table 1 depicts the basic characteristics and Table 2 presents the five-number-summaries for the #NP90k data set. While the #NP560k data set features a number of heavy users (and hence, heavy-tailed distributions), these are removed in the #NP90k data set, making it less skewed. Please note that this data set—due to its intended purpose and creation procedure—features different characteristics as the #NP560k data set.

Please note that we deliberately removed the hashtags #nowplaying, #listeningto and #listento from the data sets as at least one of those hashtags is contained in every listening event and hence do not add any further information.

In these data sets, not only listening events are tagged with hashtags, also tracks can transitively (via the listening event the track is mentioned in) be tagged with the respective hashtags. Similarly, we tag users with hashtags if a given hashtag is used within one of the listening events sent by the user. We reason that hashtags have been shown to serve two roles [40]: i) users wanting to express his/her thoughts, feelings and opinions, ii) using hashtags to tag the content of the tweet. For our study, both factors are important as we aim to evaluate the potential of affective hashtags for ranking music recommendations.

3 Evaluation Methods

In the following we present the methods deployed for the evaluation of the ranking methods presented in Section 4.

All the experiments are conducted based on the #NP560k and #NP90k data sets presented in Section 2. For conducting the evaluation, we need to split the data sets into training and test sets and apply different splitting methods for the two data sets. For #NP560k, we perform the following *per-user* split: for each user in the data set, we randomly choose 70 percent of his/her listening events as training data and the remaining 30 percent as test data. We believe that this splitting method allows for mitigating the skewness of the data set as the split is performed on a per-user bases and hence, is robust against dominating users in data set (i.e., users with a high number of listening events). In contrast, for the #NP90k data set that has already been cleaned from outliers (and is therefore less skewed), we employ a *global split* that randomly picks 70 percent out of all listening events of all users for the training set and uses the remainder of listening events as test data. These contrasting splitting approaches permit to investigate the connection between a user's emotional context and the user's concurrent musical preference independent of the size of user profiles.

For both of these splitting approaches and the underlying data sets, the latent features of nodes are computed for the items within the training set only and do not incorporate any information from the test set.

The basic input items for our evaluation are listening events, which are tweets containing information about a track a user listened to. The workflow of the evaluation is as follows. Based on a listening event randomly chosen of the test set (hereafter referred to as "input listening event") including its affective hashtags, we aim to evaluate the ranking methods proposed in Section 4.3. We consider the track contained in this input listening event as our ground truth data and our goal is to find ranking methods that rank this ground truth track first in the recommendation list.

From a recommender system point of view, our data sets represent *implicit feedback* data [16], [41]—the data sets represent traces of user behavior and they only provide us with the tracks a user has listened to. Our data set does not contain any implicit feedback by users (i.e., play counts, skipping behavior, session durations or dwell times during browsing the catalog). As most papers dealing with implicit feedback [41], what we can do is to assume that the user likes these tracks. We are not aware of the tracks that the user dislikes. In other words, all the listening events contained in our data sets are *positive* data, and there is no *negative* data at all. This has been referred to as the *one-class*

problem [42]. To learn discriminative latent feature representations, we need to perform so-called *negative sampling* [27], [28], [29] to include user-track-hashtag associations that are not present in our data sets as negative data (cf. Section 4.2). Likewise, for evaluation, we need to sample negative data to test how our ranking methods can identify the positive track and rank it on top of the list.

Different ways to perform negative sampling for the test set represents different evaluation tasks. As described in Section 1, it is possible to use the full data catalog as negative data, as many prior work on context-aware recommendation do [14], [15], [16], [17], but in this way we are not able to properly study the two RQs. Alternatively, we consider the following two evaluation tasks.

First, we aim to evaluate whether our proposed approach is able to capture the general listening preferences of users. Therefore, we propose the *POP_RND* task, where we add nine randomly chosen tracks to the list containing the input listening event to populate the list. This task allows us to evaluate whether our approach is able to capture the general listening preferences of users.

Second, we aim to evaluate a context-specific scenario where we model the sentiment of a user as the context in which tracks are listened to by users. We consider this scenario as more complex than solely capturing the general listening preferences of users. Therefore, we propose the *POP_USER* task, where we randomly pick nine tracks the user has previously listened to and add these to the set of tracks to be ranked. This requires the user to have a listening history comprising at least ten tracks.

As this task selects tracks that are associated with the user, we are able to evaluate the performance of incorporating contextual sentiment and hashtag information in the ranking computation as we have to employ context information to be able to rank those tracks effectively. Therefore, we argue that this task allows us to directly evaluate the usefulness of hashtags and sentiment scores.

We propose to evaluate the ranking performance of our approach for sets of ten tracks. In the field of recommender systems, a set of 5–10 recommendations is most appropriate which also corresponds to the capacity of short-term memory [43]. Furthermore, the work by Bollen et al. [44] underlines this choice as the authors conducted an experiments showing that presenting users with a large number of good and valuable items is counterproductive as the choice of an item becomes inherently difficult for the user.

The (unordered) set of ten tracks resulting from the proposed data generation is subsequently used as input for the recommendation ranking evaluation. In the next step, we apply the proposed ranking methods to this set of track recommendation candidates.

As for the evaluation metric, we rely on the mean reciprocal rank (MRR) metric [45] as defined in Equation (1) to evaluate the rank of the single correct item. We choose MRR as we are only interested in how the ranking methods perform in regards to ranking the ground truth track as high as possible in the ordered list of recommendation candidates. Ranking the ground truth one as the first item yields a RR of 1, ranking it second yields a RR of 0.5, etc. As the lists to be ranked in our experiments only contain a single correct item, the maximum RR obtainable is 1

TABLE 3
Sentiment Dictionaries and Their Coverage of the #NP560k Data Set; 'LE' Is a Shorthand for Listening Event

		Coverage		
Name	#Terms	Hashtags	LEs	Tracks
AFINN [32]	2,477	57.64%	46.87%	54.67%
Opinion Lexicon [33]	6,789	44.86%	44.35%	47.48%
SentiStrength [34]	2,546	71.23%	73.97%	71.50%
Vader [35]	7,517	57.63%	57.80%	61.54%

$$RR(item) = \frac{1}{rank(item)}. \quad (1)$$

In total, we repeat this evaluation procedure for a set of 20,000 listening events randomly extracted from the test set for all the proposed ranking methods and consequently, determine the MRR for the set of all ranked recommendation lists contained in the evaluated set of listening events. We use these to compare the performance of the ranking methods and the underlying latent features.

4 Computational Methods

In the following section, we present the methods utilized for leveraging affective hashtags for music recommendations.

4.1 Sentiment Detection for Hashtags

The extraction of sentiment polarity from a given word, sentence or text has been studied widely [18], [19]. Also, sentiment detection in the context of Twitter has been addressed by research [20], [21], [22]. In this study, we focus on hashtags that express emotion. Therefore, we aim to detect the sentiment of hashtags in a first step. For this task, we rely on a widely used unsupervised sentiment detection method: so-called sentiment lexica [19]. In principle, sentiment lexica are dictionaries of words, where each word is annotated with its polarity (and possibly, also the strength of this polarity). For detecting the sentiment of a term, it is simply matched against a given lexicon. In the following, we describe the specific steps taken for assigning sentiment values to the hashtags within our data set.

4.1.1 Sentiment Dictionaries

We rely on well-established dictionaries which have been widely used and evaluated [30], [31]. In particular, we use the dictionaries that provide both the best coverage and performance in terms of accuracy according to the study of Ribeiro et al. [30]. Table 3 contains an overview of the adopted lexica.

The AFINN dictionary [32] was assembled from a set of different word lists (e.g., obscene words and internet slang words) and manually annotated by a single annotator. Opinion Lexicon [33] is computed by using antonym and synonym relationships among words and using this information to deduce scores for adjectives. The SentiStrength lexicon [34] is based on a manually annotated dictionary, which is subsequently improved by adjusting the scores by machine learning techniques. The Vader dictionary [35] is also created by human annotation and is particularly geared towards sentiment analysis of social media texts.

4.1.2 Affection Computation

Based on the set of hashtags contained in the data set, we employ the following strategy to resolve hashtags against a given sentiment dictionary. Firstly, we aim to match full hashtags against the dictionary, both lowercased. However, this does only match hashtags which represent full proper English words (e.g., #happy). For all other hashtags, we apply lemmatization, as provided by the Python NLTK Wordnet package.[2] Consequently, we match these lemmata against the lemmata of the given lexicon. For hashtags that cannot be resolved directly or after lemmatization, we assume that these are either compound words or can simply not be found in the given dictionary. As for compound hashtags, these can either be written as camel case as e.g., #IAmHappy or a concatenation of multiple lowercased terms as e.g., #feelinggood. We aim to split these compound hashtags to match the single terms contained in the hashtag against the sentiment dictionaries. Therefore, we use the split words (i.e., {I, am, happy} for the above example) to represent the hashtag. As for camel case-hashtags, we split the hashtag using upper-case characters as delimiters. The problem of segmenting all-lowercase compound hashtags has already been addressed in literature [46]. Therefore, we follow previous work [47] to split these up. As the sentiment lexica are limited to English words, we base our approach on a dictionary of 109,582 English words.[3] We split the original hashtag at each position and look into whether the prefix is contained in the dictionary. If it is contained, we recursively repeat the procedure until we find an optimal result. Once we found a representation of the hashtag that consists of a set of individual terms using the methods described, we match these terms against the sentiment lexicon individually. We assign the hashtag the mean of the sentiment scores of all terms contained in the original hashtag.

Table 3 features an overview of the coverage of the different sentiment lexica. Here, we list the percentage of hashtags that can be resolved against the various dictionaries for the #NP560k data set. Similarly, we also list the fraction of listening events and tracks that can be assigned a sentiment value using the respective dictionary. Please note that despite the difference in size between the #NP560k and #NP90k data sets, the coverage of the individual sentiment lexica is comparable for both data sets and hence, we only list the coverage numbers for the #NP560k data set here. Besides using these single sentiment dictionaries, we also propose to exploit the variety and extended coverage of the combination of multiple dictionaries by using the mean value of all sentiment values across all available sentiment dictionaries gathered for a given hashtag.

The lexica have different ranges of polarity scores (e.g., AFINN from -5 to 5, and Opinion Lexicon from -1 to 1). Therefore, before computing the mean values, we normalize them by using linear min-max feature scaling.

4.2 Computation of Latent Features

While there are many methods for learning feature representations of users, tracks and hashtags from listening data, we employ the so-called network embedding technique [27], [28], [29] to learn such representations. The task of network embedding is to learn the low-dimensional representations of vertices in an information network that can capture and preserve the network structure in the representations. Such network embedding methods are useful for modeling data containing heterogeneous types, which is exactly the case here as we have users, tracks and hashtags to be modeled. In particular, we build a graph containing these three object types from the data sets and then use a network embedding algorithm to learn their representations. Although several network embedding models have been proposed, for this work we use the well-known DeepWalk approach [27]. DeepWalk is one of the most popular network embedding algorithms owing to its effectiveness in modeling the global structure of the input graph [27]. The algorithm learns low-dimensional latent feature descriptions for all the vertices (including users, tracks, and hashtags) within the graph, allowing to compute their similarity in a joint feature space.

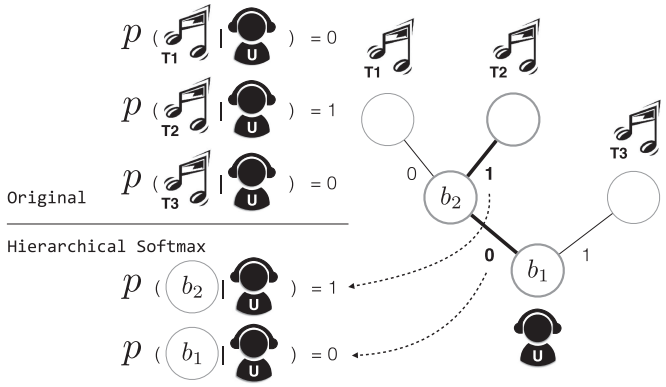


Fig. 1. Suppose the task is modeling the user-track pair (u, t_2), the original modeling function requires to compute all pair-wise estimations (i.e., $(u, t_1), (u, t_2), (u, t_3)$) while the transformed hierarchical softmax computes the estimations with only the passing nodes (i.e., $(u, b_1), (u, b_2)$).

Given a graph G and its vertices V and edges E, the objective is to model the following conditional probabilities:

$$p(v_j|v_i) = \frac{sim(v_i, v_j)}{\sum_{v_k} sim(v_i, v_k)}, \tag{2}$$

where sim is a function that measures the similarity between two vertices v_i and v_j based on their representations. Therefore, the vertices sharing similar neighbors receive similar conditional probability distribution.

To obtain the low-dimensional representations of each vertex, we further conduct a mapping function $\Phi : v \in V \mapsto \mathcal{R}^{|V| \times d}$ in Equation (2) to map the node v into a low-dimensional vector $\Phi(v)$, which also satisfies the above objective function

$$p(v_j|\Phi(v_i)) = \frac{sim(\Phi(v_i), v_j)}{\sum_{v_k} sim(\Phi(v_i), v_k)}. \tag{3}$$

Instead of computing all vertex pairs, which is quite expensive owing to the number of given vertices, DeepWalk factorizes the conditional probability using the hierarchical softmax [48] to assign each vertex a series of binary codes by Huffman tree construction. For a pair (i, j), suppose the path to vertex v_j is identified by a sequence of tree nodes $[b_0, b_1, \ldots]$, then the final objective is converted to multiple binary classification predictions:

2. http://www.nltk.org/howto/wordnet.html
3. http://www-01.sil.org/linguistics/wordlists/english/

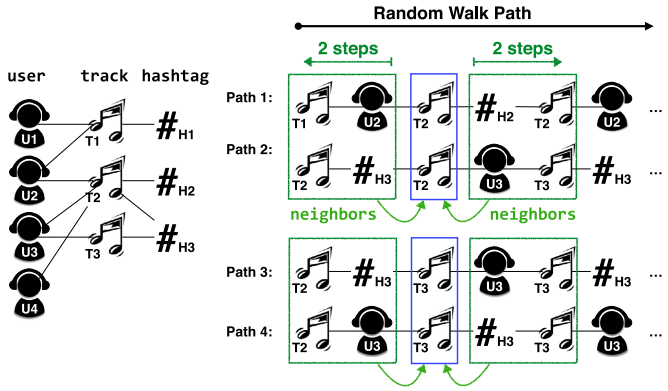


Fig. 2. The paths (right) are generated by random walks according to the given graph (left). When the window size is set to two, the connected vertices within two steps are treated as the context information of the centered vertex. In this way, vertices with similar neighbor connections will receive similar connection status and thus, receive similar probability distributions.

$$p(v_j|\Phi(v_i)) = \prod_l p(b_l|\Phi(v_i)). \quad (4)$$

Thereby, the computational complexity is reduced by the transformation from $\mathcal{O}(|V|)$ to $\mathcal{O}(\log|V|)$. Fig. 1 shows the idea of the hierarchical softmax transformation.

To further efficiently learn the low-dimensional representations, DeepWalk also uses sampling techniques for conducting the concept of random walk, which is a common technique when dealing with a huge graph. Fig. 2 plots the stochastic random walk of DeepWalk. It uses a random walk strategy to generate a path, and then adopts a certain window size to dynamically sample the observed pairs (v_i, v_j) for modeling Equation (4). The appearing probability also implies the reachability between two vertices, and can serve as sim in Equation (2). Finally, the vertices which share similar neighbors will pass similar tree node paths and thus, receive similar representations. For optimizing the representations, stochastic gradient descent [49] is utilized.

Differently designed graphs underlying the DeepWalk computation can lead to different assumptions on the relationships among the vertices in the graph.

In a conventional recommendation task, the connections between users and tracks (i.e., listening events) provide the most useful information about users' taste on music. Hence, we build a user-to-track graph (*u2t*) as the baseline network.

In our study, to analyze the impact of hashtags, we further add the connections between tracks and hashtags to the baseline network. Although there are several other ways to construct the graph, such as 'u2t2h' (i.e., no direct connection between users and hashtags), 'u2h2t' (i.e., no direct connection between users and tracks), 't2u2h' (i.e., no direct connection between tracks and hashtags), and 'uth' (i.e., allowing connections among users, tracks and hashtags), we select *u2t2h* out of the other four because in this way, the sampled random walks will always visit a track every two steps, as demonstrated in Fig. 2. According to our observations, placing the tracks at the center of the modeling process in this way obtains better representations. Consequently, we employ the following two input graphs for computing the DeepWalk latent features:

- *u2t*: This represents the user-to-track bipartite graph, the relations of which are determined by whether a user has listened to a track in previous listening events.
- *u2t2h*: The user-to-track-to-hashtag graph that further considers the links between a track and its hashtags.

4.3 Ranking

The goal of ranking is to list the most suitable items (tracks in this study) on top. It is therefore a crucial task not only in recommender systems [50], but also more generally in the area of information retrieval [51] as it directly influences precision of recommendations or search results.

The main building blocks for computing a ranking for a set of recommendation candidates are users, tracks and hashtags that are extracted from the graph. The employed network embedding technique allows us to represent users by the latent features computed for users. We refer to this representation as "*user*". To also explicitly incorporate the hashtags that a user has previously adopted into the user's representation, we propose to use the latent representations of the hashtags the user made use of, leading to the user representation "*usertags*". A user may also be represented by the average sentiment value assigned to these hashtags as a measure of the user's general sentiment, which is a scalar. We refer to this user representation as "*usersent*". Similarly, we may model a track by its latent representation in the graph ("*track*"), the latent representations of all the hashtags which have been used to tag the track ("*tracktags*"), or the average sentiment value assigned to these hashtags as the track's general sentiment ("*tracksent*"). Furthermore, we aim to exploit information about the hashtags which are used for the given input tweet by using the average latent representation of these hashtags, leading to the representation of a tweet ("*tweettags*"). Besides solely relying on latent features, we also propose to represent the input tweet as the sentiment value associated with the hashtags mentioned in the input tweet ("*tweetsent*").

Based on these building blocks, we propose the following methods for ranking a given set of tracks. In principle, these methods differ in the way users, tracks and hashtags are characterized.

- *user_track:* rank according to the similarity of the latent representations of a given user and track.
- *user_tracktags:* rank according to the average pairwise similarity of the latent representation of the user and the individual latent representations of hashtags annotating the given track.
- *usertags_track:* rank according to the average pairwise similarity of the latent representations of hashtags a user has made use of and the latent representation of the track.
- *usertags_tracktags:* rank according to the average pairwise similarity of the latent representations of the hashtags of a user and the latent representations of hashtags used for annotating the given track.
- *tweettags_track:* ranking computed based on the average pairwise similarity of the latent representations of the hashtags used in the given input tweet and the latent representation of the track to be ranked.
- *tweettags_tracktags:* rank according to the average pairwise similarity of latent representations of hashtags of

the input tweet and hashtags annotating the track to be ranked.

- *tweetsent_tracksent:* rank according to sentiment score similarity by using the difference of the sentiment scores assigned to the input tweet and the sentiment scores assigned to the tracks to be sorted. If a tweet or track features more than a hashtag, we compute the average sentiment score assigned to the track and compute the difference between these as

$$sim = abs(avg(sent(tweet)) - avg(sent(track))), \quad (5)$$

  where *sent* determines the set of sentiments assigned to a given tweet or track.
- *usersent_tracksent:* rank according to the sentiment score similarity between the average sentiment of the user's previously used hashtags and the sentiment values annotating the track.

We can use either the cosine similarity or the Euclidean distance to compute the similarity between two latent representations. This similarity score is subsequently used to actually rank the tracks in order of descending similarity.

5 Results

We conduct three experiments in our study. The first and fundamental experiment aims to verify that utilizing an embedding approach is beneficial in our setting and that embedding approaches allow to capture a user's general listening preferences. The second experiment (and for us, the central experiment) aims to extensively evaluate the performance of different ranking methods and hence, the impact of affective contextual information extracted from hashtags. The third experiment is targeted at complementing our view on sentiment-based ranking methods and investigates the performance of individual sentiment lexicon. We present the results below.

5.1 Experiment 1: Effectiveness of Latent Features

In the first experiment, we aim to show that incorporating latent features contributes to a better ranking, capturing the general listening preferences of users. We therefore base this evaluation on the POP_RND task and evaluate the performance of the user_track ranking method (similarity of latent features of users and tracks), where latent features are computed based on the user-to-track graph (u2t). Hence, we do not consider any hashtag or affective information in this first experiment. We compare this approach with the following baseline methods:

- A random ranking approach that randomly shuffles the items within the recommendation list;
- Ranking according to the tracks' popularity within our data set (i.e., the number of distinct users having listened to the track) [52], [53]. Picking random items or the most popular items are basic and simple baselines often used for dealing with the cold-start problem [52];
- An item-item-based collaborative filtering approach based on the k-nearest neighbors (kNN) [54]. We set the size of the neighborhood k to 30 and use cosine similarity to measure the similarity between items, following the suggestion of Sarwar et al. [54]. Herlocker et al. have also found that generally, a neighborhood size of 20 to 50 seems reasonable for real-world settings [55].

TABLE 4
The Mean Reciprocal Rank (MRR) Achieved by Different Ranking Methods for POP_RND for Both the #NP90k and #NP560k Data Sets (Standard Deviation in Parentheses)

Ranking Method	#NP560k	#NP90k
Random	0.29 (0.26)	0.29 (0.26)
Most popular tracks	0.73 (0.32)	0.76 (0.30)
kNN	0.81 (0.33)	0.79 (0.35)
user_track (u2t embedding; cos.)	**0.92** (0.21)	**0.81** (0.34)
user_track (u2t embedding; eucl.)	0.83 (0.31)	0.68 (0.41)

There are a few parameters to be empirically decided for the DeepWalk algorithm for learning the latent features. In a preliminary study we found that the following setting works reasonably well: dimension of the latent representation, which controls the model complexity—64, number of walks and the walk length, which control the number of sampling pairs for the modeling stage—16 and 64 respectively, the window size, which determines the reachable vertices—4. We use this parameter setting throughout the following experiments, for both u2t and u2t2h.

The results of the conducted analysis are listed in Table 4. As can be seen, incorporating latent features increases the quality of the ranking compared to the baseline methods. The random baseline reaches an average MRR of 0.29 for both data sets, while ranking according to the popularity of tracks reaches a MRR of 0.73 (#NP560k data set) and 0.76 (#NP90k data set). Among the baselines, the item-item collaborative filtering baseline (kNN) reaches a MRR 0.81 (#NP560k data set) and 0.79 (#NP90k data set), respectively.

The use of latent representations of tracks and users increases the MRR to 0.92 for #NP560k and 0.81 for #NP90k. Also, cosine similarly outperforms euclidean similarity. Generally, from this first experiment we conclude that incorporating latent features in the ranking process yields improved results compared to the evaluated baseline approaches. Hence, this validates the effectiveness of the latent features for capturing a user's general musical preferences.

5.2 Experiment 2: Effectiveness of Affection and Hashtag Information

The goal of this experiment is to examine the benefit of incorporating hashtag and affective information into the ranking process. Ultimately, we aim to evaluate the performance of the individual proposed ranking strategies in a context-aware ranking task. Therefore, we consider both the POP_RND and POP_USER task in this experiment.

Table 5 depicts the results of this evaluation for the #NP560k data set and Table 6 presents the results for the #NP90k data set. As our experiments showed that cosine similarity consistently outperforms Euclidean similarity by a small margin, we only list the results of cosine similarity. For POP_RND, we see that the best results are obtained by the user_track ranking method, achieving a MRR of 0.92 (u2t embedding; #NP560k data set) and 0.83 (u2t2h embedding; #NP90k data set). For the #NP90k data set, usertags_track also reaches a MRR of 0.83. As for the user_track ranking method, we do not observe substantial differences between ranking

TABLE 5
The MRR Achieved by Different Ranking Methods, Using Cosine Similarity for the #NP560k Data Set (Standard Deviation in Parentheses)

Ranking method	Graph	POP_RND	POP_USER
user_track	u2t	**0.92** (0.21)	0.28 (0.26)
user_track	u2t2h	0.91 (0.22)	0.29 (0.26)
user_tracktags	u2t2h	0.88 (0.26)	0.80 (0.32)
usertags_track	u2t2h	0.89 (0.24)	0.29 (0.26)
usertags_tracktags	u2t2h	0.84 (0.29)	0.78 (0.32)
tweettags_track	u2t2h	0.89 (0.23)	0.32 (0.29)
tweettags_tracktags	u2t2h	0.86 (0.27)	0.80 (0.31)
tweetsent_tracksent	—	0.81 (0.34)	**0.82** (0.30)
usersent_tracksent	—	0.39 (0.30)	0.68 (0.32)

TABLE 6
The MRR Achieved by Different Ranking Methods, Using Cosine Similarity for the #NP90k Data Set (Standard Deviation in Parentheses)

Ranking method	Graph	POP_RND	POP_USER
user_track	u2t	0.81 (0.34)	0.22 (0.22)
user_track	u2t2h	**0.83** (0.32)	0.22 (0.21)
user_tracktags	u2t2h	0.79 (0.33)	0.56 (0.37)
usertags_track	u2t2h	**0.83** (0.31)	0.25 (0.23)
usertags_tracktags	u2t2h	0.74 (0.34)	0.59 (0.37)
tweettags_track	u2t2h	0.84 (0.30)	0.36 (0.33)
tweettags_tracktags	u2t2h	0.78 (0.34)	0.65 (0.37)
tweetsent_tracksent	—	0.70 (0.37)	**0.71** (0.35)
usersent_tracksent	—	0.42 (0.32)	0.53 (0.32)

approaches incorporating hashtags (i.e., u2t2h) and those not incorporating hashtags (i.e., u2t) in the latent features representation. As for the other ranking approaches, we observe that usersent_tracksent, user_tracktags, tweettags_tracktags and usertags_tracktags reach lower MRR values. Notably, the tweetsent_tracksent ranking method, which solely relies on the sentiment scores associated with the tracks to be ranked and hashtags the user made use of in the current input tweet, achieves 0.81 (#NP560k data set) and 0.70 (#NP90k data set).

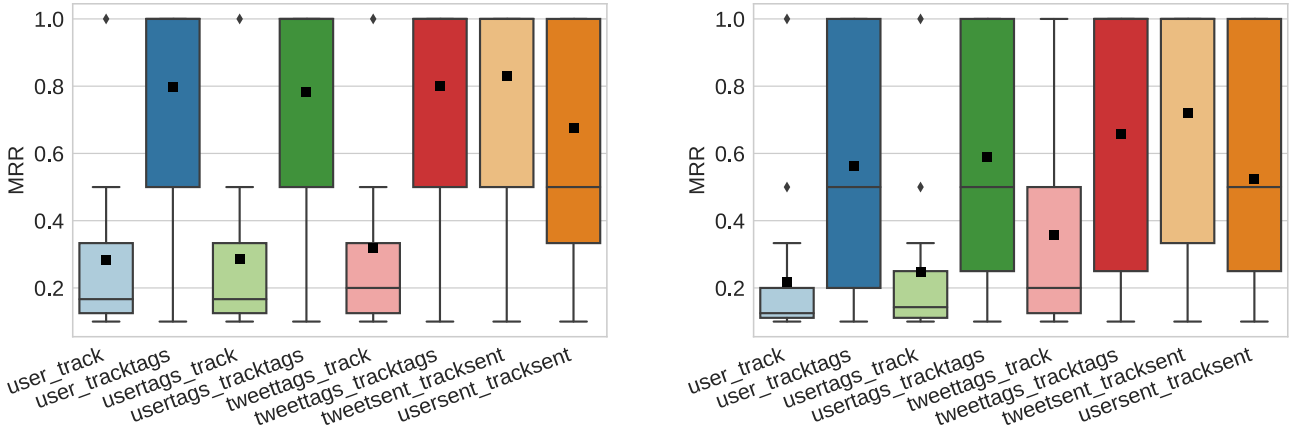
In contrast, for POP_USER, we can see from Tables 5 and 6 that the sentiment ranking method tweetsent_tracksent outperforms all other methods, achieving the highest MRRs of 0.82 (#NP560k data set) and 0.71 (#NP90k data set), respectively. The results support our hypothesis that sentiment hashtags and embeddings incorporating hashtags allow for better capturing a user's context and hence, exploiting this information for ranking track candidates. For a better comparison, we also provide a boxplot of the MRR for both data sets in Fig. 3. The other sentiment-based ranking method, usersent_tracksent, achieves a MRR of 0.68 and 0.53, respectively. Notably, these methods do not use latent features. Methods utilizing "tracktags" for representing tracks, including user_tracktags, tweettags_tracktags, and usertags_tracktags, also perform well and reach a MRR around 0.80 for #NP560k and 0.60 for #NP90k. In contrast, the user_track method performs poorly here, with a MRR below 0.30 across all settings. In general, methods using "track" for representing tracks do not perform well. These findings suggest that contextual affective information and in general, information about the tags used to describe tweets or tracks is indeed exploited in this task. This is also signaled by the fact that methods that incorporate latent features of hashtags and sentiment information perform substantially better than the approach not incorporating such information.

In sum, we argue that ranking tracks the user has already listened to is more challenging than ranking a set of randomly chosen tracks as these traditionally differ more. Therefore, we consider this result as promising. Our experiments also show that both data sets (and hence, splitting methods regarding training and test data) deliver robust and consistent results.

5.3 Experiment 3: Effectiveness of Individual Sentiment Lexica

In this experiment we aim to get a deeper understanding for the performance of different sentiment detection approaches or rather, lexica. Therefore, we now focus on the performance of the sentiment-aware ranking methods and firstly evaluate the performance of single sentiment lexica.

The usage of sentiment dictionaries for the detection of sentiment in a text is naturally limited by the coverage of the given sentiment dictionary (cf. Section 4.1 regarding the coverage of the sentiment lexica used). This limited coverage consequently constrains the number of affective hashtags detectable



(a) #NP560k Data set (b) #NP90k Data set

Fig. 3. Boxplot of MRR achieved by different ranking methods using POP_USER (u2t2h embedding; black square marks mean value across all evaluations).

TABLE 7
Performance (in MRR) of Different Sentiment Dictionaries for tweetsent_tracksent in the #NP560 Data Set (Standard Deviation in Parentheses)

Dictionary	Fallback	POP_RND	POP_USER
AFINN	None	0.79 (0.34)	0.79 (0.34)
Opinion Lexicon	None	0.81 (0.32)	0.80 (0.33)
SentiStrength	None	0.85 (0.29)	0.85 (0.27)
Vader	None	0.87 (0.25)	0.85 (0.29)
AFINN	user_track	0.85 (0.29)	0.81 (0.31)
Opinion Lexicon	user_track	0.86 (0.28)	0.82 (0.31)
SentiStrength	user_track	0.86 (0.28)	0.84 (0.29)
Vader	user_track	0.89 (0.24)	0.85 (0.28)

TABLE 8
Performance (in MRR) of Different Sentiment Dictionaries for tweetsent_tracksent in the #NP90 Data Set (Standard Deviation in Parentheses)

Dictionary	Fallback	POP_RND	POP_USER
AFINN	None	0.68 (0.39)	0.68 (0.37)
Opinion Lex.	None	0.71 (0.38)	0.69 (0.37)
SentiStrength	None	0.72 (0.36)	0.73 (0.35)
Vader	None	0.77 (0.33)	0.77(0.34)
AFINN	tweettags_tracktags	0.77 (0.35)	0.68 (0.36)
Opinion Lex.	tweettags_tracktags	0.79 (0.33)	0.68 (0.36)
SentiStrength	tweettags_tracktags	0.76 (0.35)	0.73 (0.34)
Vader	tweettags_tracktags	0.80 (0.22)	0.74 (0.34)

using any single dictionary which further limits the number of tracks which can be actually assigned with a sentiment score. Thus, only a limited number of tracks can be compared in this regard.

To compare the different lexica nonetheless, we propose the following method. For those tracks, users and tweets for which we can compute a sentiment score using the given dictionary, we rely on the best performing ranking method tweetsent_tracksent as evaluated in the previous experiments. However, for the remaining tracks, users and tweets with no sentiment scores, we employ a *fallback* method. Here we distinguish two cases: i) if we cannot detect a sentiment score for either the user or the tweet, we use the fallback method for all the tracks to be ranked; ii) if we cannot detect a sentiment for a track (or a set thereof), we compute the similarity of user (or tweet; depending on the ranking method) and the track using the fallback method. As for the fallback methods, we chose to use and evaluate the best-performing ranking methods not relying on affective information for each task. Hence, we evaluate user_track for POP_RND and tweettags_tracktags for POP_USER as fallback methods, respectively and utilize the average sentiment score detected for a given tweet or track for the comparison.

Tables 7 and 8 show the results for the #NP560 and #NP90k data set, respectively. Here, we consider POP_RND as a special case as the best performing method is not sentiment-based and the user_track fallback method performs better than the sentiment-aware ranking methods. Hence, the usage of such a fallback method naturally increases the performance of the evaluation where the degree of improvement depends on the coverage of the dictionary used. However, the goal of this evaluation is to evaluate the individual dictionaries and therefore, we still list the results. To provide a complete picture of the results, we also list the performance of the individual sentiment dictionaries when no fallback method is used (i.e., 'Fallback None'). As the table shows, the best results (by a slight margin) are obtained using the user_track fallback method. Examining the dictionaries used, we do observe slight differences but note that Vader performs the best. As for POP_USER, we observe that in this case, using no fallback method performs slightly better than using the fallback method as our experiments in Section 5.2 already showed that tweetsent_tracksent is the best performing ranking strategy (again, by a moderate margin). As for the individual dictionaries, we find that the differences in regards to the MRR are rather moderate with Vader again performing the best for both the user- and the tweet-based sentiment ranking.

6 Discussion

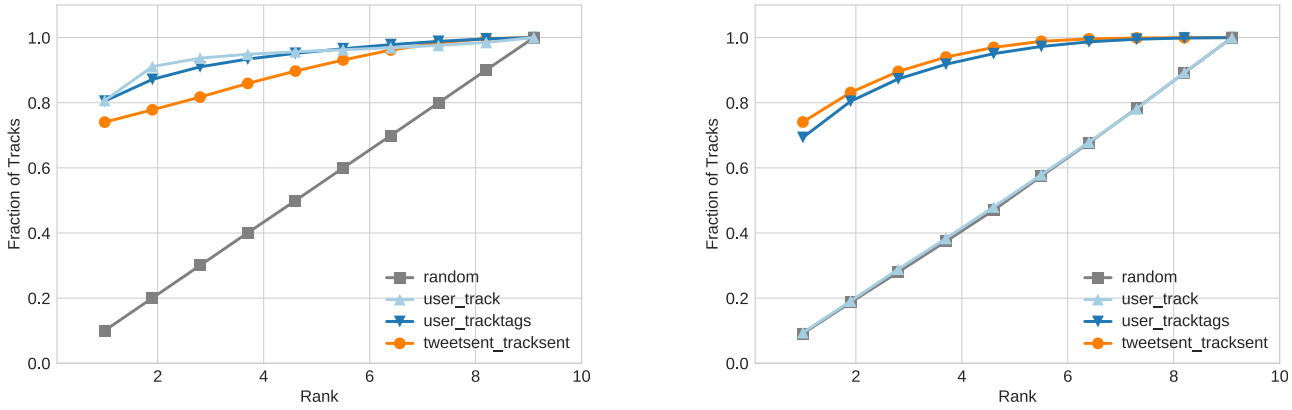
We further discuss the evaluation results in this section.

In the first experiment we showed that representing tweets, tracks and users by latent features computed by the DeepWalk algorithm and using similarities between these for ranking tracks achieves comparable results as traditional ranking methods. Therefore, we conclude that these latent representations are able to capture users' general listening preferences and that those can be used for ranking tracks in a recommendation or retrieval scenario. Our experiment comparing the performance of user_track learned from u2t and u2t2h sees marginal differences in terms of MRR, for either POP_RND or POP_USER. In this scenario, these findings signal that hashtag information integrated in the computation of latent features does hardly influence the resulting latent feature representations for users and tracks.

However, using u2t2h as the underlying graph permits learning the latent feature representations for hashtags, which is useful for the POP_USER task. That is, for the context-aware ranking task, hashtags (providing contextual information) naturally contribute to an improved ranking. Analyzing the results of the different proposed ranking methods in our second experiment, we find that using the latent representations of hashtags that are used to tag tracks (i.e., "tracktags") seem to be more representative of a track than using solely the latent representation of the track itself (i.e., "track") for POP_USER. We can observe that tracktags performs substantially better over all configurations. However, this does not hold for POP_RND. These findings suggest that for POP_RND, the latent representation of a track seems more suitable than using the hashtags annotating a track. This shows that for capturing a user's general listening preferences, utilizing the latent representations of users and tracks are sufficient for computing a suitable ranking.

Similarly, users can either be represented by the user's latent feature representation ("user") or by the latent representations of the hashtags the user made use of ("usertags"). However, we encounter mild differences between the performance of these two representations for either POP_RND or POP_USER. Hence, we conclude that the differences of different representations for users are hardly distinctive.

Among the two sentiment-based ranking methods, we find that using the sentiment of the input tweet ("tweetsent") performs better for both POP_RND and POP_USER. These



(a) POP_RND (b) POP_USER

Fig. 4. Cumulative ranking distribution of different methods for the #NP560k data set.

results suggest that using the sentiment expressed by the user in the current tweet captures the current affective context better than using the average sentiment a user has previously expressed through hashtags. This can also be seen in Fig. 4, which plots the cumulative ranking function for the random baseline and the ranking methods user_track, user_tracktags, tweetsent_tracksent (utilizing the average score across all sentiment lexica). For POP_RND we observe that user_track provides superior results across all ranks incorporated. In contrast, for POP_USER we observe that user_track shows behavior highly similar to the random ranking approach (those two lines actually overlap heavily), whereas tweetsent_tracksent and user_tracktags perform substantially better across all ranks.

From these experiments we observe that choosing a suitable representation for tracks, users and tweets is crucial for the quality of the ranking. We find that for POP_RND, comparing the latent representations of users and tracks is sufficient to provide high-quality ranking of tracks. However, the POP_USER experiment showed that this does not suffice when the ranking task gets more personal and complex. This experiment showed that ranking based on contextual affective information performs best. Particularly, the tweetsent_tracksent ranking method outperformed the other methods. From these findings we conclude that while for the POP_RND task the latent representations did capture the user's preference well, for the POP_USER task the sentiment did capture the user's musical interest better.

The third experiment aimed to evaluate the performance of the individual sentiment lexica and hence, their suitability for this task. We observed that Vader performed best across all evaluations. However, we have to note that the differences are rather moderate. Given that Vader performs similar to the other dictionaries in terms of coverage, we lead this back to the fact that Vader is a particularly geared towards social media texts.

Our evaluation design proposes a fallback method to compensate for those tweets, users and tracks which could not be assigned with a sentiment score using the given dictionary. This naturally implies that the choice of the fallback ranking method is vital. We propose to evaluate the best performing algorithm not considering sentiment data. For POP_RND, the fallback methods individually perform better than the sentiment-based ranking methods. Thus, an improvement of the results when introducing user_track is an obvious result. The tweettags_tracktags ranking method is also able to improve the results, though to a lower degree. As already laid out, we consider the POP_USER task as the more difficult and personal task. For this evaluation, results worsened by the fallback methods as expected, since these methods did not perform as well as the sentiment-based methods in the previous experiments. From these results we reason that implementing a fallback method is a good choice as it provides means for compensating the lack of coverage. Also, we conclude that choosing the fallback method according to the complexity and degree of how personal the ranking task is, seems plausible. As for the choice of sentiment dictionaries, we propose to employ the union of multiple dictionaries to increase coverage. While our experiments show that minor improvements for single dictionaries, we argue that in this case, coverage should be prioritized as it allows for a higher applicability of the sentiment-based ranking, which has shown to perform better.

We also aim to acknowledge the limitations of the work presented. Relying on sentiment dictionaries for detecting the sentiment of hashtags is a rather naive approach. While we believe that this method is sufficiently elaborate for the experiments conducted, there are a number of shortcomings. Firstly, the approach is highly dependent on the underlying dictionaries and their coverage (as shown in our experimental results). Second, matching hashtags (either full hashtags or parts thereof) against sentiment dictionaries is agnostic to linguistic aspects such as adverbs that describe the extent and strength of emotions (such as in #verysad) or negations (such as #notfunny). We aim to extend and enhance the sentiment detection method utilized in future work by utilizing state-of-the-art approaches for sentiment detection. Regarding the computation of rankings, some of our methods rely on computing the mean values of e.g., sentiment scores detecting in listening events of a user. This approach is insensitive to high variance in the underlying data. Particularly, for users with high variance in their sentiment scores, taking the average of these scores dilutes this highly useful information. Therefore, we aim to look into more sophisticated approaches to represent such distributions using e.g., probabilistic models such as Gaussian mixture models [56] in our future work.

7 Background and Related Work

Before concluding the paper, we give a brief review of related work in psychology and recommender systems, to put this work in the context of the literature.

7.1 Psychological Studies on Emotion Regulation

Emotion regulation is important for the performance and well-being of mankind [57]. It is widely accepted that emotions play a major role in driving our decisions. Beneficial emotion regulation strategies help people to stay calm under stress, handle failures in a mindful and positive way, etc. Due to its importance, emotion regulation has been recognized as one of the fastest growing areas within the field of psychology [58], [59].

Emotion regulation has also been identified as an essential reason for musical engagement [3], [60], [61]. Boer and Fischer [62] found that emotion regulation represents the most important personal use of music across human subjects from four cultural backgrounds. Goethem and Sloboda [3] found that music listening is the second-most used tactic for emotion regulation, just behind "talking with friends".[4]

Saarikallio et al. [63] found that a person's general tendency to emotionally appreciate, enjoy and react to music (i.e., *emotional reactivity to music*) is positively correlated with the tendency to use music for emotion regulation (i.e., *emotional use of music*). Being familiar with a music piece increases a person's emotional use of that piece in daily life [63]. Moreover, informal engagement through listening, but not formal musical training, correlate with heightened emotional use of music. Saarikallio also argued that music should not simply be considered as one emotion regulation mechanism, but rather as a tool for realizing several different emotion regulation strategies, including positive mood maintenance, relaxation and revival, induction of strong emotions, diverting away from worries, discharging negative emotion, mentally working through emotion preoccupations, and finding solace and understanding [64]. Individual differences in the use of these strategies have also been noted: e.g., some prefer emotional reinforcement of current experiences, while others prefer to distract themselves and change emotions [65], [66].

While many psychological studies were conducted in the lab with small to medium sample size, what we investigate here is the relationship between a user's self-report emotional state and the self-report musical preference through Twitter at scale and "in the field." Moreover, a computational approach that investigates how to represent the affective information of users and music using machine learning and sentiment detection techniques is taken. Although our study may also lead to psychological insights, the focus is more on the engineering side, targeting at applications such as affective music recommendation.

There have been psychological evidences showing that the emotional state of users affects musical preference. For example, depressed patients expressed an intensified response to sad-sounding music when compared to healthy controls [67]. Moreover, such patients evaluated negative-valence music as significantly more sad and angry than healthy controls did [68]. However, according to Gross [57], people do not always attempt to stay away from negative emotions. Reasons for up-regulating negative emotions include promoting a focused, analytic mindset; fostering an emphatic stance; and influencing others' actions.

An interesting research direction is therefore to use Twitter data to computationally study the effect of music in emotion regulation at scale using a longitudinal approach. This requires tracking specific users' #nowplaying tweets and emotional states over time, which to our knowledge has not been attempted before. We leave this as a future work.

Finally, we remark that Hargreaves and North [60] proposed that music has three types of psychological functions: cognitive, emotional, and social functions. The focus of this paper is on the emotional functions of music, neglecting the possible social functions manifested in the Twitter data.

7.2 Affective Multimedia Recommendation

Contextual factors relevant to music recommendation may include the time, location and device of music listening, user's present emotional state and activity, etc. [69]. While it is relatively easier to infer some of these factors from sensors such as clocks, GPS and accelerometers [70], [71], [72], accessing the emotional state of a user is more difficult. As users may not always be willing to report their emotions, computational methods for user emotion prediction from facial expressions, prosody cues, text, and physiological signals have been widely studied [73], [74], [75].

With 40K blog posts collected from the social blogging website LiveJournal,[5] Yang and Liu [76] investigated the relationship between the emotional state of a user and the emotion of preferred music pieces. Similar to the Twitter data set we use in this paper, the LiveJournal data set they employed also contains the self-report emotional states and self-report preferred music pieces [77]. Yang and Liu [76] used audio signal processing and machine learning techniques to recognize the emotion of the music [78] and then correlated the emotion in music with the emotional state of the users, finding that users do prefer music of different emotions in different emotional states. Following this work, Chen et al. [26] showed that considering the emotional state of a user indeed improves the quality of music recommendation, comparing to conventional collaborative filtering approaches that do not use affective information. This article extends from these two prior articles in that we use a larger data set and more sentiment detection methods.

Ferwerda and Schedl [6] proposed the idea of exploiting both the personality and emotional state of a user for music recommendation, but did not actually implement such a system. Rosa et al. [7] built a system that recommends music according to the emotional states inferred from user-generated text by sentiment detection, but the system was evaluated using a small-scale data set collected from a crowdsourcing platform, not from social media websites. Deng et al. [23] assumed that the emotional state of a user can be determined by the emotions of the music pieces the user just listened to. There are some other affective music recommender systems proposed in the literature, but many of them require users to indicate their present emotional states or the desired emotions of the music [24], [25], [79].

4. The other tactics considered in their study include "exercising", "reading a book/magazine", "watching TV/movie", among others [3].

5. http://www.livejournal.com

Affective movie recommendation has also been studied in recent years, using mainly the users' self-report emotional states [80], [81], [82]. Although it might be possible to crawl movie preference data from social platforms such as Twitter, few attempts have been made thus far.

8 Conclusion and Future Work

In this paper, we have proposed a set of novel methods for ranking music recommendation candidates. In particular, we proposed to represent the building blocks (users, tracks, affective hashtags) by their latent features computed by a network embedding algorithm called DeepWalk. Based on these latent feature representations, we proposed a number of ranking methods. Furthermore, we proposed two ranking methods that are solely based on sentiment scores. Our evaluation using #nowplaying tweets showed that the use of latent features to represent users, tracks and hashtags contributes to better ranking. The evaluation procedure distinguished two tasks of increasing complexity: i) ranking a set of randomly picked tracks and ii) ranking a set of tracks the target user has already listened to. We find that for the first task, comparing the latent representations of users and tracks (regardless of used hashtags or affective information) performs best and this confirms our hypothesis that applying an embedding technique effectively captures the general listening preferences of users. However, for the second, context-aware ranking task, using solely affective information extracted from hashtags leads to the best result. We consider the second task is a more complex and, from a user-perspective, more personal ranking problem. Our findings suggest that in this case, contextual, affective information is able to better capture the user's preference. Finally, an evaluation of the different sentiment lexica showed that the differences in performance of the individual lexica is rather moderate. While Vader achieves the best results, we argue that combining several dictionaries or implementing fallback methods results in a more robust approach.

Future work includes incorporating more sophisticated sentiment detection approaches both regarding the underlying dictionaries as well as the computation of the sentiment scores. In a first step, we aim to further evaluate different aggregation methods for tracks that are tagged with multiple tags with divergent sentiment scores (e.g., `#happysad`). Furthermore, we aim to find common characteristics of users or tracks regarding their ranking performance (e.g., to look into the performance of the proposed approach for users that have a high variance in sentiment scores). Also, we aim to experiment with probabilistic models for representing a user's or track's sentiment values (e.g., using Gaussian Mixture Models [56]).

Modeling affective information in a multidimensional model such as the valence-arousal space [78], [83] is worth exploring. Also, we aim to extend the unsupervised sentiment-detection approach (based on sentiment dictionaries) to a supervised learning approach that permits cross-lingual sentiment detection [84]. Lastly, the computation of latent features based on the proposed graph needs to be investigated in more detail. Particularly, we aim to investigate the influence and performance of different embedding strategies for the computation of latent representations. Ultimately, we intend the development and evaluation of real-world applications for music recommendation and music-based emotion regulation based on our findings.

Acknowledgments

The computational results presented have been achieved (in part) using the HPC infrastructure LEO of the University of Innsbruck. The research of Chen, Tsai and Yang is supported by a grant from the Ministry of Science and Technology, Taiwan, under project MOST 106-3114-E-002-007.

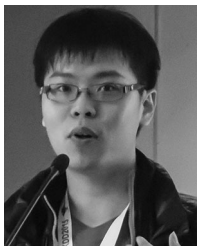
References

[1] T. Schäfer, P. Sedlmeier, C. Stdtler, and D. Huron, "The psychological functions of music listening," *Frontiers Psychology*, vol. 4, no. 511, pp. 1–34, 2013.
[2] A. J. Lonsdale and A. C. North, "Why do we listen to music? A uses and gratifications analysis," *Brit. J. Psychology*, vol. 102, pp. 108–134, 2011.
[3] A. Van Goethem and J. A. Sloboda, "The functions of music for affect regulation," *Musicae Scientiae*, vol. 15, no. 2, pp. 208–228, 2011.
[4] M. E. Sachs, A. Damasio, and A. Habibi, "The pleasures of sad music: A systematic review," *Frontiers Human Neurosci.*, vol. 9, no. 404, pp. 1–12, 2015.
[5] L. Baltrunas, M. Kaminskas, B. Ludwig, O. Moling, F. Ricci, A. Aydin, K.-H. Lke, and R. Schwaiger, "InCarMusic: Context-aware music recommendations in a car," in *E-Commerce and Web Technologies*, C. Huemer and T. Setzer, Eds. Berlin, Germany: Springer, 2011, pp. 89–100.
[6] B. Ferwerda and M. Schedl, "Enhancing music recommender systems with personality information and emotional states: A proposal," in *Proc. Conf. User Model. Adaptation Personalization*, 2014, pp. 36–44.
[7] R. L. Rosa, D. Z. Rodrguez, and G. Bressan, "Music recommendation system based on user's sentiments extracted from social networks," *IEEE Trans. Consum. Electron.*, vol. 61, no. 3, pp. 359–367, Aug. 2015.
[8] D. Watson and R. Mandryk, "An in-situ study of real-life listening context," in *Proc. Sound Music Comput. Conf.*, 2012, pp. 11–16.
[9] E. Zangerle, M. Pichl, W. Gassler, and G. Specht, "#nowplaying music dataset: Extracting listening behavior from Twitter," in *Proc. Int. Workshop Internet-Scale Multimedia Manage.*, 2014, pp. 21–26.
[10] M. Pichl, E. Zangerle, and G. Specht, "#nowplaying on #Spotify: Leveraging Spotify information on Twitter for artist recommendations," in *Proc. Workshop Current Trends Web Eng.*, 2015, pp. 163–174.
[11] D. Hauger, M. Schedl, A. Košir, and M. Tkalčič, "The million musical tweets dataset: What can we learn from microblogs," in *Proc. Int. Soc. Music Inf. Retrieval Conf.*, 2013, pp. 189–194.
[12] M. Schedl, "The LFM-1b Dataset for music retrieval and recommendation," in *Proc. ACM Int. Conf. Multimedia Retrieval*, 2016, pp. 103–110.
[13] G. Adomavicius, B. Mobasher, F. Ricci, and A. Tuzhilin, "Context-aware recommender systems," *AI Mag.*, vol. 32, no. 3, pp. 67–80, 2011.
[14] A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver, "Multiverse recommendation: N-dimensional tensor factorization for context-aware collaborative filtering," in *Proc. ACM Conf. Recommender Syst.*, 2010, pp. 79–86.
[15] Y. Shi, M. Larson, and A. Hanjalic, "Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges," *ACM Comput. Survey*, vol. 47, no. 1, pp. 3:1–3:45, 2014.
[16] S. Rendle, "Factorization machines with libFM," *ACM Trans. Intell. Syst. Technol.*, vol. 3, no. 3, pp. 57:1–57:22, 2012.
[17] T. V. Nguyen, A. Karatzoglou, and L. Baltrunas, "Gaussian process factorization machines for context-aware recommendations," in *Proc. ACM Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2014, pp. 63–72.
[18] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retrieval*, vol. 2, no. 1/2, pp. 1–135, 2008.
[19] A. Neviarouskaya, H. Prendinger, and M. Ishizuka, "SentiFul: A lexicon for sentiment analysis," *IEEE Trans. Affect. Comput.*, vol. 2, no. 1, pp. 22–36, Jan.-Mar. 2011.

[20] E. Kouloumpis, T. Wilson, and J. Moore, "Twitter sentiment analysis: The good the bad and the OMG!" in *Proc. Int. AAAI Conf. Weblogs Social Media*, 2011, pp. 538–541.
[21] P. Nakov, Z. Kozareva, A. Ritter, S. Rosenthal, and V. S. T. Wilson, "SemEval-2013 task 2: Sentiment analysis in Twitter," in *Proc. 2nd Joint Conf. Lexical Comput. Semantics*, 2013.
[22] A. Severyn and A. Moschitti, "Twitter sentiment analysis with deep convolutional neural networks," in *Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2015, pp. 959–962.
[23] J. J. Deng, C. H. C. Leung, A. Milani, and L. Chen, "Emotional states associated with music: Classification, prediction of changes, and consideration in recommendation," *ACM Trans. Interactive Intell. Syst.*, vol. 5, no. 1, pp. 4:1–4:36, 2015.
[24] B.-J. Han, S. Rho, S. Jun, and E. Hwang, "Music emotion classification and context-based music recommendation," *Multimedia Tools Appl.*, vol. 47, no. 3, pp. 433–460, 2010.
[25] M. Barthet, D. Marston, C. Baume, G. Fazekas, and M. B. Sandler, "Design and evaluation of semantic mood models for music recommendation using editorial tags," in *Proc. Int. Soc. Music Inf. Retrieval Conf.*, 2013, pp. 421–426.
[26] C.-M. Chen, M.-F. Tsai, J.-Y. Liu, and Y.-H. Yang, "Using emotional context from article for contextual music recommendation," in *Proc. ACM Int. Conf. Multimedia*, 2013, pp. 649–652.
[27] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 701–710.
[28] A. Grover and J. Leskovec, "Node2Vec: Scalable feature learning for networks," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2016, pp. 855–864.
[29] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "LINE: Large-scale information network embedding," in *Proc. Int. Conf. World Wide Web*, 2015, pp. 1067–1077.
[30] F. N. Ribeiro, M. Araújo, P. Gonçalves, F. Benevenuto, and M. A. Gonçalves, "SentiBench - a benchmark comparison of state-of-the-practice sentiment analysis methods," *EPJ Data Sci.*, vol. 5, no. 1, p. 23, 2016, doi: 10.1140/epjds/s13688-016-0085-1.
[31] P. Gonçalves, M. Araújo, F. Benevenuto, and M. Cha, "Comparing and combining sentiment analysis methods," in *Proc. ACM Conf. Online Social Netw.*, 2013, pp. 27–38.
[32] F. Å. Nielsen, "A new ANEW: Evaluation of a word list for sentiment analysis in microblogs," in *Proc. ESWC2011 Workshop 'Making Sense Microposts': Big Things Come Small Packages*, 2011, pp. 93–98.
[33] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2004, pp. 168–177.
[34] M. Thelwall, K. Buckley, G. Paltoglou, D. Cai, and A. Kappas, "Sentiment strength detection in short informal text," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 61, no. 12, pp. 2544–2558, 2010.
[35] C. J. Hutto and E. Gilbert, "VADER: A parsimonious rule-based model for sentiment analysis of social media text," in *Proc. Int. Conf. Weblogs Social Media*, 2014, pp. 216–225.
[36] Twitter: Twitter Filter API. [Online]. Available: https://dev.twitter.com/docs/api/1.1/post/statuses/filter
[37] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a social network or a news media?" in *Proc. ACM Int. Conf. World Wide Web*, 2010, pp. 591–600.
[38] K. Lerman and R. Ghosh, "Information contagion: An empirical study of the spread of news on Digg and Twitter social networks," in *Proc. 4th Int. Conf. Weblogs Social Media*, 2010, pp. 90–97.
[39] D. M. Romero, W. Galuba, S. Asur, and B. A. Huberman, "Influence and passivity in social media," in *Proc. 20th Int. Conf. Companion World Wide Web*, 2011, pp. 113–114.
[40] L. Yang, T. Sun, M. Zhang, and Q. Mei, "We know what @you #tag: Does the dual role affect hashtag adoption?" in *Proc. ACM Int. Conf. World Wide Web*, 2012, pp. 261–270.
[41] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *Proc. IEEE Int. Conf. Data Mining*, 2008, pp. 263–272.
[42] R. Pan, Y. Zhou, B. Cao, N. N. Liu, R. M. Lukose, M. Scholz, and Q. Yang, "One-class collaborative filtering," in *Proc. IEEE Int. Conf. Multimedia Expo.*, 2008, pp. 502–511.
[43] G. A. Miller, "The magical number seven, plus or minus two: Some limits on our capacity for processing information," *Psychological Rev.*, vol. 63, no. 2, 1956, Art. no. 81.
[44] D. Bollen, B. P. Knijnenburg, M. C. Willemsen, and M. Graus, "Understanding choice overload in recommender systems," in *Proc. 4th ACM Conf. Recommender Syst.*, 2010, pp. 63–70.
[45] E. M. Voorhees et al., "The TREC-8 question answering track report," in *Proc. 8th Text Retrieval Conf.*, 1999, pp. 77–82.
[46] S. Srinivasan, S. Bhattacharya, and R. Chakraborty, "Segmenting web-domains and hashtags using length specific models," in *Proc. ACM Int. Conf. Inf. Knowl. Manage.*, 2012, pp. 1113–1122.
[47] O. Tsur and A. Rappoport, "What's in a hashtag?: Content based prediction of the spread of ideas in microblogging communities," in *Proc. ACM Int. Conf. Web Search Data Mining*, 2012, pp. 643–652.
[48] A. Mnih and G. Hinton, "A scalable hierarchical distributed language model," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2008, pp. 1081–1088.
[49] L. Bottou, "Stochastic gradient learning in neural networks," *Proc. Neuro-Nımes*, vol. 91, no. 8, 1991.
[50] G. Adomavicius and Y. Kwon, "Improving aggregate recommendation diversity using ranking-based techniques," *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 5, pp. 896–911, May 2012.
[51] S.-T. Park and D. M. Pennock, "Applying collaborative filtering techniques to movie search for better ranking and browsing," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2007, pp. 550–559.
[52] A. M. Rashid, I. Albert, D. Cosley, S. K. Lam, S. M. McNee, J. A. Konstan, and J. Riedl, "Getting to know you: Learning new user preferences in recommender systems," in *Proc. Int. Conf. Intell. User Interfaces*, 2002, pp. 127–134.
[53] Z. Gantner, S. Rendle, C. Freudenthaler, and L. Schmidt-Thieme, "MyMediaLite: A free recommender system library," in *Proc. ACM Conf. Recommender Syst.*, 2011, pp. 305–308.
[54] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. Int. Conf. World Wide Web*, 2001, pp. 285–295.
[55] J. Herlocker, J. A. Konstan, and J. Riedl, "An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms," *Inf. Retrieval*, vol. 5, no. 4, pp. 287–310, 2002.
[56] J.-C. Wang, Y.-H. Yang, H.-M. Wang, and S.-K. Jeng, "The acoustic emotion Gaussians model for emotion-based music annotation and retrieval," in *Proc. ACM Int. Conf. Multimedia*, 2012, pp. 89–98.
[57] J. Gross, "Emotion regulation: Conceptual and empirical foundations," in *Handbook of Emotion Regulation*, 2nd ed., J. Gross, Ed. New York, NY, USA: The Guilford Press, 2007, pp. 1–19.
[58] S. L. Koole, "The psychology of emotion regulation: An integrative review," *Cognition Emotion*, vol. 23, pp. 4–41, 2009.
[59] M. Tamir, "The maturing field of emotion regulation," *Emotion Rev.*, vol. 3, pp. 3–7, 2011.
[60] D. J. Hargreaves and A. C. North, "The functions of music in everyday life: Redefining the social in music psychology," *Psychology Music*, vol. 27, no. 1, pp. 71–83, 1999.
[61] P. N. Juslin and P. Laukka, "Expression, perception, and induction of musical emotions: A review and a questionnaire study of everyday listening," *J. New Music Res.*, vol. 33, no. 3, pp. 217–238, 2004.
[62] D. Boer and R. Fischer, "Towards a holistic model of functions of music listening across cultures: A culturally decentred qualitative approach," *Psychology Music*, vol. 40, no. 2, pp. 179–200, 2010.
[63] S. Saarikallio, S. Nieminen, and E. Brattico, "Affective reactions to musical stimuli reflect emotional use of music in everyday life," *Musicae Scientiae*, vol. 17, no. 1, pp. 27–39, 2013.
[64] S. Saarikallio, "Music in mood regulation: Initial scale development," *Musicae Scientiae*, vol. 12, pp. 291–309, 2008.
[65] S. Saarikallio, "Music-related emotional self-regulation across adulthood years," *Psychology Music*, vol. 39, pp. 307–328, 2011.
[66] M. Tamir, "Don't worry, be happy? Neuroticism, trait-consistent affect regulation and performance," *J. Personality Social Psychology*, vol. 89, no. 3, pp. 449–461, 2005.
[67] E. Bodner, I. Iancu, A. Gilboa, A. Sarel, A. Mazor, and D. Amir, "Finding words for emotions: The reactions of patients with major depressive disorder towards various musical excerpts," *Arts Psychotherapy*, vol. 34, no. 2, pp. 142–150, 2007.
[68] M. Punkanen, T. Eerola, and J. Erkkilä, "Biased emotional recognition in depression: Perception of emotions in music by depressed patients," *J. Affect. Disorders*, vol. 130, no. 1/2, pp. 118–126, 2011.
[69] P. Knees and M. Schedl, "Music retrieval and recommendation: A tutorial overview," in *Proc. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2015, pp. 1133–1136.

[70] Y.-H. Yang and Y.-C. Teng, "Quantitative study of music listening behavior in a smartphone context," *ACM Trans. Interactive Intell. Syst.*, vol. 5, no. 3, 2015, Art. no. 14.
[71] X. Wang, D. Rosenblum, and Y. Wang, "Context-aware mobile music recommendation for daily activities," in *Proc. ACM Int. Conf. Multimedia*, 2012, pp. 99–108.
[72] M. Schedl, G. Breitschopf, and B. Ionescu, "Mobile music genius: Reggae at the beach, metal on a Friday night?" in *Proc. ACM Int. Conf. Multimedia Retrieval*, 2014, pp. 507–510.
[73] R. A. Calvo and S. D'Mello, "Affect detection: An interdisciplinary review of models, methods, and their applications," *IEEE Trans. Affect. Comput.*, vol. 1, no. 1, pp. 18–37, Jan.-Jun. 2010.
[74] Z. Zeng, M. Pantic, G. I. Roisman, and T. S. Huang, "A survey of affect recognition methods: Audio, visual, and spontaneous expressions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 1, pp. 39–58, Jan. 2009.
[75] Y.-P. Lin, Y.-H. Yang, and T.-P. Jung, "Fusion of electroencephalogram dynamics and musical contents for estimating emotional responses in music listening," *Frontiers Neurosci.*, vol. 8, no. 94, pp. 1–14, 2014.
[76] Y.-H. Yang and J.-Y. Liu, "Quantitative study of music listening behavior in a social and affective context," *IEEE Trans. Multimedia*, vol. 15, no. 6, pp. 1304–1315, Oct. 2013.
[77] J.-Y. Liu, S.-Y. Liu, and Y.-H. Yang, "LJ2M dataset: Toward better understanding of music listening behavior and user mood," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2014, pp. 1–6.
[78] Y.-H. Yang and H.-H. Chen, "Machine recognition of music emotion: A review," *ACM Trans. Intell. Syst. Technol.*, vol. 3, no. 3, 2012, Art. no. 40.
[79] I. Andjelkovic, D. Parra, and J. O'Donovan, "Moodplay: Interactive mood-based music discovery and recommendation," in *Proc. ACM Conf. User Model. Adaptation Personalization*, 2016, pp. 275–279.
[80] Y. Shi, M. Larson, and A. Hanjalic, "Mining mood-specific movie similarity with matrix factorization for context-aware recommendation," in *Proc. Workshop Context-Aware Movie Recommendation*, 2010, pp. 34–40.
[81] Y. Zheng, B. Mobasher, and R. D. Burke, "The role of emotions in context-aware recommendation," in *Proc. RecSys Workshop Human Decision Making Recommender Syst.*, 2013, pp. 21–28.
[82] A. K. M. Tkalčič and J. Tasič, "Affective recommender systems: The role of emotions in recommender systems," in *Proc. RecSys Workshop Human Decision Making Recommender Syst.*, 2011, pp. 9–13.
[83] J. Wang, L. C. Yu, K. R. Lai, and X. Zhang, "Community-based weighted graph model for valence-arousal prediction of affective words," *IEEE/ACM Trans. Audio Speech Language Process.*, vol. 24, no. 11, pp. 1957–1968, Nov. 2016.
[84] C. R. Argueta, F. H. Calderon, and Y.-S. Chen, "Multilingual emotion classifier using unsupervised pattern extraction from microblog data," *Intell. Data Anal.*, vol. 20, no. 6, pp. 1477–1502, 2016.

Eva Zangerle received the master's degree in computer science and the PhD degree in recommender systems for collaborative social media platforms, both from the University of Innsbruck. She is a postdoctoral researcher with the University of Innsbruck at the research group for Databases and Information Systems (Department of Computer Science). Her main research interests include within the fields of social media analysis, recommender systems, and information retrieval. Over the last years, she has combined these three fields of research and investigated music recommender systems based on data retrieved from social media platforms aiming to exploit new sources of information for recommender systems. She was awarded a postdoctoral fellowship for overseas researchers from the Japan Society for the Promotion of Science allowing her to make a short-term research stay at the Ritsumeikan University in Kyoto.



Chih-Ming Chen is working toward the PhD degree in the Taiwan International Graduate Program of the Academia Sinica and the Department of Social Networks and Human-Centered Computing, National Chengchi University. He has also been a research intern at KKBOX, a leading music streaming company in East Asia, and is in charge of developing the music recommendations toolkit. Moreover, he has participated and won many public data competitions such as KDD Cup, so he is experienced in both predictive modeling and data analysis. His research interests span the territories of machine learning and recommender system.

Ming-Feng Tsai received the PhD degree from the National Taiwan University, in 2009. He is currently an associate professor with the Department of Computer Science, National Chengchi University. In 2006, he was at Microsoft Research Asia as a visiting student with the Web Search & Mining Group, and was awarded by the research institution the "Best Intern of the Year." After receiving his PhD degree, he worked with the National University of Singapore as a research fellow, participating in a research project related to machine translation. In 2010, sponsored by Taiwan National Science Council, he joined the University of Illinois at Urbana-Champaign as a visiting scientist, working on a project associated with advanced Web search and mining. His research interests span the areas of information retrieval, machine learning, natural language processing, and recommender systems. In 2014, he served as financial chair of the International Society for Music Information Retrieval Conference (ISMIR).

Yi-Hsuan Yang (M'11–SM'17) received the PhD degree in communication engineering from the National Taiwan University, in 2010. He is an associate research fellow with Academia Sinica. He is also a Joint-Appointment associate professor with the National Cheng Kung University, Taiwan. His research interests include music information retrieval, affective computing, multimedia, and machine learning. He was a recipient of the 2011 IEEE Signal Processing Society Young Author Best Paper Award, the 2012 ACM Multimedia Grand Challenge First Prize, the 2014 Ta-You Wu Memorial Research Award of the Ministry of Science and Technology, Taiwan, and the 2015 Best Conference Paper Award of the IEEE Multimedia Communications Technical Committee. He is an author of the book *Music Emotion Recognition* (CRC Press 2011). In 2014, he served as a technical program co-chair of the International Society for Music Information Retrieval Conference (ISMIR). In 2016, he started his term as an associate editor of the *IEEE Transactions on Affective Computing* and the *IEEE Transactions on Multimedia*. He is a senior member of the IEEE.