Framework for Emotional Context Aware Music Recommendation System

Arjun K
Dept. of Computer Science and Engineering
National Institute of Technology
Tiruchirappalli, India
arjun.k018@outlook.com

Dr. E. Sivasankar
Dept. of Computer Science and Engineering
National Institute of Technology
Tiruchirappalli, India
sivasankar@nitt.edu

***Abstract*— There has been significant importance to context aware recommendations in the recent years, especially when users are faced with a large number of contents for them to choose from. The present widespread use of micro-blogging serves as a platform for users to share their emotions. There has been research intended to use the user emotions shared in micro-blogging platforms to perform music recommendations. But with the present scenario, the amount of data available is huge and the computation time of recommendations are increasing rapidly. The paper intends to provide a method by which to improve the computation cost and the scalability of recommendations without compromising on accuracy and provide near to realtime music recommendations for users on basis of mood of user obtained from micro-blogging platforms.**

Keywords—Recommender system; Context aware recommendations;

I. INTRODUCTION

The recent years have seen a surge in the amount of digital music content available in users due to the increased penetration of internet in various parts of the world. This calls out for the need to search for recommender systems for users to be able to find the music based on their interest easily. The past decade has seen an increase of more than 10 times in the number of digital music content available for download. There are various websites that provide these content for free or for a small cost. Most of these web stores have their own recommender systems to facilitate easier discovery of music by their users. They use context based filtering or collaborative filtering approach for music recommendations by taking into considerations music, genre, artist and various other features. The main point that is lacking in this approach is the inability to tailor the recommendations to the context of users like location, mood, weather, location, etc. Various works exist that makes use of various context information of users like weather(Park, Yoo, & Cho, 2006), location (Cheng & Shen, 2014;Kaminskas, Ricci, & Schedl, 2013; Schedl, Vall, & Farrahi, 2014), emotion(Shuiguang Deng, Dongjing Wang, Xitong Li, Guandong Xu, 2015). Most of these contexts can be directly inferred from data about users – location (gps coordinates), weather (weather data). The contexts like emotion cannot be directly inferred from data from users. These types of contexts are called secondary contexts and they can be inferred from various indirect methods like analysis of the content in blogs of users in various platforms. Music preference depends highly on the emotional context of the user. It is challenging to obtain emotion context from indirect methods and apply it in a recommender system and compute recommendations in a reasonable time. Most of the music web stores have a huge number of users and music content. The task of computing recommendations for this huge amount of data is a challenging. This calls for optimizing various databases so that we can improve the scalability of the recommender system without any compromise on the accuracy of the system. Present works do not focus on the scalability of emotion aware recommender systems.

In this paper we propose a recommender system that uses the emotional context of users taken from microblogging platforms like Twitter and provide music recommendations. Due to increase in the penetration of smart phones and internet, social networking platforms have become an essential part of lives of people. Publishing microblogs have become an essential part of life of people and they share their emotions in these microblog platforms. The recommender system proposed in the paper is based on the assumption that the posts of users in these microblog platforms reflects the emotion of the user at that time. The main scalability in the system is when there are millions of users and millions of songs, the recommender system will have to traverse huge amounts of data and by the time the system comes up with recommendations the context of users would have changed. To accommodate this change there should exist a database that contains only a subset of users, so that the system will only need to traverse a small amount of data to compute recommendations. We use the LJ2M (Live Journal Two Million) dataset (http://mac.citi.sinica.edu.tw/LJ#.WCbvD3V948o) that is available for free in the internet to create the dataset of users and music for recommendations. To obtain microblog posts of users we use the Twitter Api. Twitter is a popular micro-blogging platform.

II. RELATED WORK

Fang-Fei Kuo, Meng-Fen Chiang, Man-Kwan Shan, Suh-Yin Lee [1] proposed a novel model for emotion based music recommendations. Shuiguang Deng, Dongjing Wang, Han-Saem Park, Ji-Oh Yoo, Sung-Bae Cho [2] created a context aware music recommendation system using Bayesian Networks and Utility Theory. Xitong Li, Guandong Xu, [3] proposed a method which had very good accuracy. Most of the work in this topic is novel.

III. MOTIVATION

This work is motivated by the desire to create a scalable system for the recommendation of music based on the emotional context of user. While implementing an existing research on context aware music recommendation, it was found that by the time the recommendations are retrieved for the system, the user mood had changed. This paper is intended to address this issue. The article also propose a system to obtain the feedback of users to improve recommendations. By this paper we hope to implement a framework and hope to see emotion aware music recommendations in near future.

IV. OVERVIEW OF RECOMMENDATION SYSTEM

The system that is developed consists of four phases. The first phase is the preprocessing where the database of the most valid users and music is obtained. Then the emotional analysis part. Here the most recent microblog post of the user is taken and a database of emotionally similar users are obtained from the initial dataset. The music predication phase, the emotionally similar users are taken and music recommendations are provided based on these user's historic data. In the feedback phase the initial database is modified to accommodate new data.

A. Preprocessing

The main aim of the preprocessing phase is to get the database for easier computation of recommendations. The LJ2M dataset is taken and manipulated to get the required dataset. The resulting dataset has the following tuples (user,emotion,music,not_refer_count), user field contains the user id of the particular user, emotion is a tuple containing the emotional context in which the user listened to the particular music piece, music is the name of music that is referenced by the user in the given emotional context, not_refer_count is the number of times the given tuple failed to reach the main dataset. It is a ten element tuple where each element the intensity of particular emotion ('anger', 'anticipation', 'disgust', 'fear', 'joy', 'sadness', 'surprise', 'trust', 'positive', 'negative'). The emotions 'positive' and 'negative' describes the higher granularity emotion and this is given a slightly more weight when computing the similarity. The dataset that is obtained in this phase is the most important component of the recommender system as all the other parts rely on this dataset for computation.

B. Emotion analysis

In this phase the most recent post of the user is taken and is subjected to emotional analysis. This used to find the various similar users from the main dataset. Each post is represented as an emotional vector with dimensions similar to the emotion vector in the main dataset. Based on the similarities between the emotion vector of the post and emotion vectors in the main dataset, we form association data that is used for further computation of recommendations.

C. Music Prediction Phase

In this phase the emotional vector that is obtained from the post is taken and similarities are computed against the main dataset. From the similarities a dataset of top similar users are obtained. From this dataset music-music similarity is computed using the main dataset. The number of similar users can be set manually. The parameter number of users, show a tradeoff between accuracy and speed. The smaller the number, the faster will be the algorithm. The higher the number, the accuracy of recommendations is increased. This parameter can be tuned according to requirements. The music is ranked according to the user's interest value and the top 'n' music can be recommended for the user. Here 'n' depends on the practical application of the recommender system.

D. Feedback Phase

This phase provides the flexibility that is required in the recommender system. The user is analyzed and the music that the user listens to is tracked. Then the emotional analysis phase output is taken and is added to the initial dataset. The preprocessing phase can be run repeatedly occasionally to update the initial dataset with new music and users. An AB testing framework can also be incorporated into the system to analyze the prediction accuracy of the recommender system. When the feedback phase is completed and the resulting dataset becomes larger than a preset maximum value, the additional data is eliminated by using the 'not_refer_count' attribute of the dataset. The ones with the highest 'not_refer_count' is removed from the dataset.

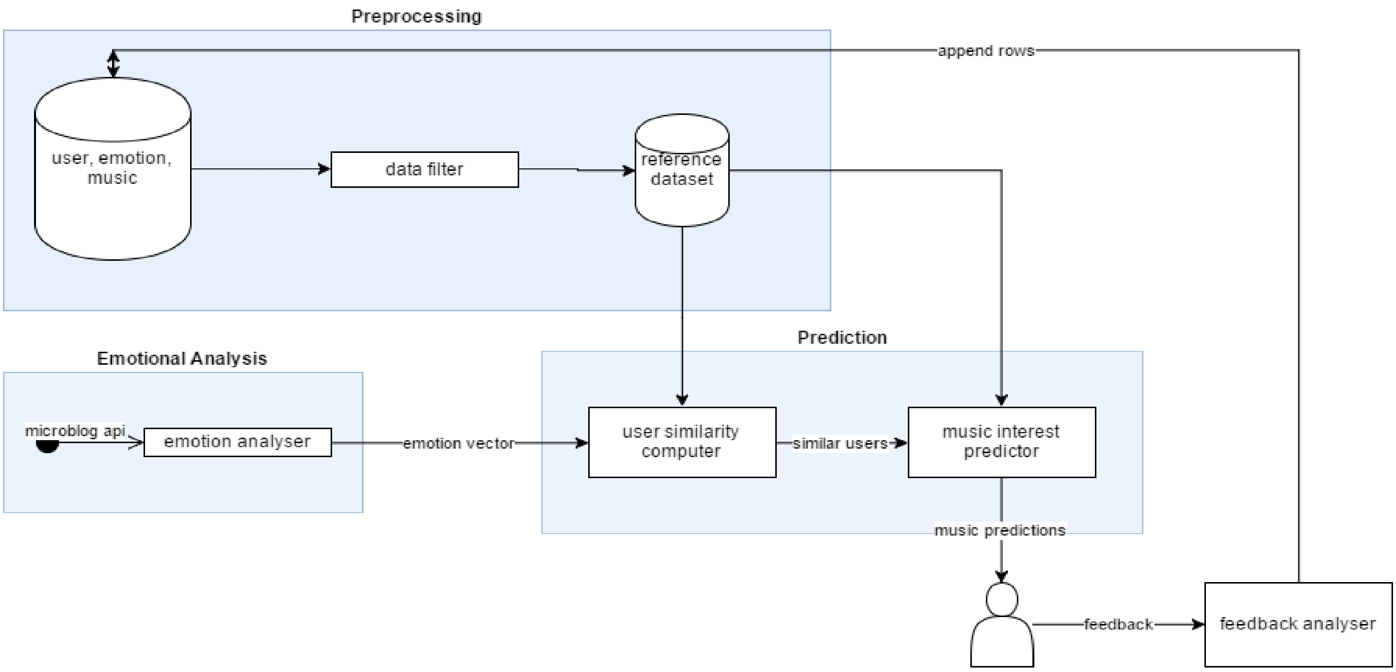
V. PREPROCESSING

Initially in this phase we use the LJ2M dataset to generate the initial data. LJ2M dataset has the following files and folders that are useful.

- ./music_emotion_tag_in_anew.txt – contains the user emotion tags in the ./info files.
- ./info/ – contains entries of LiveJournal grouped by user moods. Each file in the folder is of the form<emotiontag>.csv where "emotiontag" if from the 'music_emotion_tag_in_anew.txt' file Each entry is of the form of following tuple:

(<mood>_<article ID>,<user ID>,<original text in music tag>,<artist>,<song title>,<7digital song ID>,<EchoNest track ID>)

Each emotion in the 'music_emotion_tag_in_anew.txt' file is taken and the 10-D emotion vector is found using a emotional lex file that was obtained from the internet that contains emotional tags for different words. The resulting dataset consists of 1,398,169 entries. This dataset is first shuffled to remove uniformity in the dataset and then is divided into train and test datasets. The optimum split is found to be at 20% test data and 80% train data. Each entry in the test data is taken and similar users are found in the train data. This operation is time consuming, finally we achieve a dataset that consists of tuples of the following form:



(<user_reference>,<emotion_reference>,<target_user>,<similarity>)

- user_reference – user in the test dataset
- emotion_reference – emotion corresponding to user in test-data
- target_user – the user in train dataset that is similar to the reference_user
- similarity – similarity score of the particular target_user

The users are sorted in descending order of similarity scores and the top n (parameter that can be set according to application) users are taken for a particular user and added to a user_similarity dataset. The similarity is found by the following formula.

$$sim(x,y) = \frac{\sum_{i \varepsilon I_x \cap I_y} cos'(e_{xi}, e_{yi})}{\sqrt{|I_x|\,|I_y|}}$$

where,

- x, is target user in test dataset, y is the user in train dataset
- I_x - music set listened to by x
- I_y - music set listened to by y
- e_{xi} - emotional context of x for music i
- e_{yi} - emotional context of y for music

$$cos'(x,y) = cos(x_{v_1}, y_{v_1}) + \alpha \times cos(x_{v_2}, y_{v_2})$$

- α - adaptability coefficient for lower granularity emotions - ‘positive’ and ‘negative’
- v_1 – v_1 vector of the corresponding emotion vector
- v_2 – v_2 vector of the corresponding emotion vector

The dataset that is obtained consists of top similar users for each user in the test dataset. Now top n(depends on trade-off required between performance and accuracy) users that appear the most number of times in the top users of each user in the train set is taken. This is the main dataset that will be used for further computations in the recommendations.

VI. Emotion Analysis

This part of the recommender system is used for mining the emotion of user from the most recent post of from a microblog. For the purpose of our experiment we used Twitter Api and programmed it using a python library named Tweepy to follow individual users. Whenever the user tweets, the recommender system is activated, computing the music recommendations. We used the Twitter stream api to get the tweets of users. Suppose the users are part of a music web store. The website can be programmed to follow the tweets of the particular user to provide music recommendations. To perform emotion analysis on tweets, the tweets are considered as bag of words and using the emotional lexicon dataset, that was used for creation of main dataset, a 10 dimensional vector is created. For example consider the following tweet -

"I mean I'm frequently disgruntled. And I only have my own account!"

This tweet will be converted to the following vector -

(1,0,1,0,0,1,0,1,0,1)

This vector corresponds to values assigned to various emotional words in the tweet.

(<anger>, <anticipation>, <disgust>, <fear>, <joy>, <sadness>, <surprise>, <trust>, <positive>, <negative>)

This vector is further used in the process to find user-user similarities and then assign scores to various music pieces according to predicted user preference.

VII. Prediction Phase

This is the part of the system that is involved in the active prediction of music recommendations based on emotional context of users. The dataset derived from the preprocessing phase is used for prediction. The dataset is created in such a way that it has only some number of entries that is optimized for performance in a given system. For our experiment we fixed the number of entries in the dataset to 5,000. The 10 dimensional vector in the emotion analysis phase is taken and broken down into two separate vectors – V_1 whose dimension is 8 and V_2 whose dimension is 2. V_1 corresponds to the emotions anger, anticipation, disgust, fear, joy, sadness, surprise, trust and V_2 corresponds to the emotions positive and negative. These two vectors help to define the different granularities with which we assign different emotions to tweets. The prediction phase is divided into two separate parts.

A. *User Similarity*

From the main dataset, top users that are similar to the target user and emotional context is taken. Then we use a collaborative filtering approach to find similarity among users. The collaborative filtering weights all users in the dataset according to the similarity of emotional context based on historic music listening data. Two user pairs *x,y* are evaluated using the following equation.

$$sim(x,y) = \frac{\sum_{i \varepsilon I_x \cap I_y} cos'(e_{xi}, e_{yi})}{\sqrt{|I_x|\,|I_y|}}$$

where,

- x, is target user in test dataset, y is the user in train dataset
- I_x- music set listened to by x
- I_y- music set listened to by y
- e_{xi}- emotional context of x for music i
- e_{yi}- emotional context of y for music

$$cos'(x,y) = (cos(x_{v_1}, y_{v_1}) + \alpha \times cos(x_{v_2}, y_{v_2}))$$

- α - adaptability coefficient for lower granularity emotions - 'positive' and 'negative'
- v_1 – v_1 vector of the corresponding emotion vector
- v_2 – v_2 vector of the corresponding emotion vector

A particular number of users are selected for further computation according to the performance of system. We took the top 50 similar users for further computation.

B. *Music Ranking*

In this part of the prediction phase we consider the present emotional context of user to assign weights to various music pieces according to the predicted user interest to the particular piece. The ranks are computed using a modified similarity matrix using the collaborative filtering approach.

$$p(u,i) = \sum_{v \epsilon U_{u,k} \cap U_i} (simi(u,v) \times cos'(e_u, e_{vi}))$$

where,

- *p(u,i)* – interest value of a music piece i for user u
- *simi(u,v)* – similarity value that is computed in user similarity part of the prediction phase
- e_u – present emotional context of user u
- e_{vi} – emotional context of user v when listening to music piece i
- $U_{u,k}$ – top k similar users similar to user u
- U_i – set of all users that listened to music i

and,

$$cos'(x,y) = cos(x_{v_1}, y_{v_1}) + \alpha \times cos(x_{v_2}, y_{v_2})$$

- α – lower granularity adaptability coefficient.
- v_1 – v_1 vector of the corresponding emotion vector
- v_2 – v_2 vector of the corresponding emotion vector.

Using the above we get the music pieces that are listened to by similar users in similar emotional context. The resulting dataset is sorted in descending order of predicted interest value of the particular music piece.

VIII. Feedback Phase

This is the phase that is involved in updating the dataset with new and updated music pieces and users. After the prediction phase is complete, the users are given an option for feedback for recommendations. If a positive feedback is obtained, the corresponding top users and music pieces are added to the initial dataset. A maximum value is specified for the number of entries in the base dataset. If the resulting dataset exceeds the maximum value, some entries are purged according to the not_refer_count which contains the number of times the entry failed to reach the main dataset. The entries with the maximum not_refer_count is removed first. This helps to purse outdated music and users that do not contribute to the recommendations of the system, improving performance. This phase helps to scale the system to a large number of users.

IX. Running Environment

A. *Software Environment*

Operatin System – Ubuntu 16.04-64bit
Programming Language – Python
Packages used – Pandas, Numpy, Tweepy

B. *Hardware Environment*

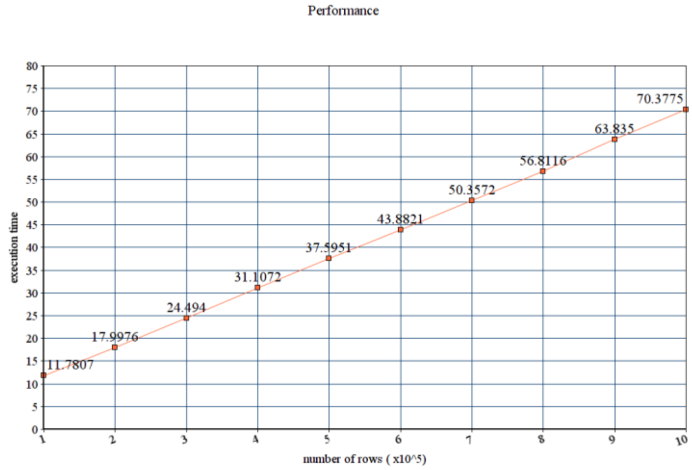
Processor – Intel Corei5
Main Memory – 4 GB
Secondary Memory – 500 GB SATA

C. *Recommendation System Statistics*

Number of users – 157796
Number of music pieces – 28573
Number of entries in initial dataset - 1398170

X. Experimental Analysis

In the experimental setup, we first computed music recommendations for the users in the test dataset derived from the main dataset. It is found that the accuracy of the system increases when we increase the number of users in our reference dataset but at the cost of time. We took an ideal value for the time requirement, 60 seconds, and found that this occurred when the number of rows in the dataset is 800000. Now this system is taken in real world and applied. Twitter data of random users are analyzed. The complete execution of the recommender system pipeline required an average of 56.8116 seconds for computation. These values are a great improvement compare to the previous method wherein the complete data of the main dataset is taken for computation. Initial system, the number of entries in the dataset were 2000000 and the average execution time prediction is found out to be 130.3756s.



XI. Conclusion

The main objective of this paper was the creation of a recommendation framework that is scalable and gives emotional context aware recommendations. From the experiments we were able to complete the system that has good performance for recommendations in practical applications. From this work it can be understood that emotional context aware recommendations can be used in real life scenarios for commercial purposes. Previous works in emotional context aware recommender systems were focused on the accuracy of the system. Here we explore the performance and accuracy of such a system to make it useful for practical purposes. However the following limitations are observed in the work:

- Only music name is considered for recommendations, whereas in practical use lot of other attributes like genre, artist, year, etc. are present.
- We consider only the emotional context for recommendations. Other contexts like location, weather should also be considered.
- The approach taken for process of emotional analysis is a naive. New techniques should be used to analyze user post to create an accurate emotional vector for a tweet.

XII. Future Work

There are multiple possibilities for future works to improve the system. One is the integration of this system to an online music service like (Google Play Music, Spotify, Pandora) and provide recommendations for registered users based on the emotional context of users. We will also try to improve the overall prediction accuracy by using the feedback phase integrated with the online music service. We will try to improve the prediction of current emotional context of users by using advanced approach for sentiment analysis of text. User level sentiment analysis incorporating social networks[4] or we will use deep learning structures like CNN-LSTM (Convoluted Neural Network-Long Short Term Memory)[5] which can predict emotional vectors from text corpus with very high accuracy. We will try to improve the recommendation performance by assessing more features of music like genre, artist, tempo, pitch profile and also create online feedback experiments to improve the accuracy of recommendations.

REFERENCES

[1] Fang-Fei Kuo, Meng-Fen Chiang, Man-Swan Shan, Suh-Yin Lee, Emotion-based music recommendation by association discovery from film music, MULTIMEDIA `05 Proceedings of the 13th annual ACM international conference on Multimedia Pages 507-510

[2] A Context-Aware Music Recommendation System Using Fuzzy Bayesian Networks with Utility Theory, Han-Saem Park, Ji-Oh Yoo, Sung-Bae Cho, Lipo Wang, Licheng Jiao, Guanming Shi, Xue Li, Jing Liu, Fuzzy Systems and Knowledge Discovery: Third International Conference, FKSD 2006, Xi'an, China, September 24-28,2006. Proceedings.

[3] Shuiguang Deng, Dongjing Wang, Xitong Li c, Guandong Xu, Exploring user emotion in microblogs for music recommendation, Expert System with Applications: An International Journal, Volume 42, Issue 23, Decemeber 2015

[4] Chenhao Tan, Lillian Lee, Jie Tang, Long Jiang, Ming Zhou, PingLi, User-level sentiment analysis incorporating social networks. KDD `11 Proceeding of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. Pages 1397-1405

[5] Jin Wang, Liang-Chih Yu, K. Robert Lai and Xuejie Zhang, Dimensional Sentiment analysis using Regional CNN-LSTM. DOI: 10.18653/v1/P16-2037, Conference: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)