# Music Recommendation based on Text Mining

Ziwon Hyung[1], Kyogu Lee[1,2]
[1]*Music and Audio Research Group*
[2]*Advanced Institutes of Convergence Technology*
*Seoul National University*
*Seoul, Korea*
*Email: ziotoss@gmail.com, kglee@snu.ac.kr*

MyoungA Lee
*High Technology Division*
*Lotte Data Communication Co.*
*Seoul, Korea*
*Email: maeng111@gmail.com*

***Abstract*—Recommending music from millions of items is a challenging problem. In this paper, we propose a novel approach to recommending music given an textual input from the user. To this end, we first mine a large corpus of textual documents from the radio station's Internet bulletin board. Each document, written by a listener, contains a personal story associated with a song request. Assuming that the personal story contains the reason for the song request, we then perform the Latent Semantic Analysis (LSA) on these documents to find the document similarity, which we believe also indicates similar music preference. Our hypothesis is that when the two users request the same song, the situation or context in which they write the associated story is likely to be similar as well, and therefore the two stories will also be similar to each other. Using the mined documents that request the same song as a test set, we show that there is a positive correlation between the document similarity and song similarity, and thus it is possible to recommend music purely based on text mining and analysis.**

***Keywords-text mining; Latent Semantic Analysis; music recommendation.***

## I. Introduction

Rapid growth in the volume of digital music data raised issues in selecting which music the user would like to listen to. This phenomenon, so-called the Paradox of Choice [13], shows that as the number of options grow, the effort in making a wise selection also increases, resulting in the selection process being a burden. Therefore, recommendation systems are becoming increasingly important due to their ability to filter out the unnecessary or unimportant data from the huge growing volume of accessible data [3]. While there are numerous approaches in music recommendation system, they can be generalized into two categories depending on how they retrieve new items: (1) collaborative filtering based recommender and (2) content-based recommender.

Collaborative filtering based music recommender identifies similar users or items based on prior purchase history and rating to recommend new items. An important requirement for this approach is that the selected item must have enough valid information provided by the users. As a consequence, it is prone to the so-called Cold Start problem [12], which with high probability misses the newly arrived items due to lack of information. Another problem is that the diversity of the recommended item is poor [17]. This problem can be explained by a phenomenon known as Long Tail [2]. Huge concentration of users is focused on popular items while only a small amount demand other items. According to the Digital Music Report 2012 [4], the combined sales of the top ten digital singles marked about 86.2 million copies. Considering the total amount of digital music sold, this number is significant. This indicates that the music industry follows the Long Tail phenomenon. Since collaborative filtering method is based on the preference of users as a criterion for recommendation, this results in using only a small portion of the music data when recommending new items.

Content-based filtering music recommender uses metadata such as genre, artist, and lyrics [9], [10], [16], and/or acoustic features [7], [8] to find similar items. While this approach is immune to the cold start problem and popularity bias of the CF approach, it faces other issues such as computational power. Since the music database is extremely large and still expanding, using content-based recommendation approach requires a huge amount of time to analyze the content and recommend similar music, and thus it is inefficient for commercial use. Another problem is that the system must be provided with an input music in order to compare the content and provide a recommendation list. This again leads to the cold start problem and also the paradox of choice.

While CF method and content-based method have its own issues, a common problem in both approach is that they neglect an important criterion; the user's situational information when one seeks to listen to music. Recently, people tend to write their daily situational information via social network services. From this observation, we thought of using such textual information to extract the contextual information when recommending music. The idea of our algorithm is to perform Latent Semantic Analysis (LSA) [6] on the documents retrieved from the radio station's Internet bulletin board to discover similar stories. The audience of the radio channel writes their own story in the bulletin board and requests a song they would like to listen to as a consequence of the story. In this paper, we will use the term document

 ISBN: 978-1-61208-227-1 

to indicate the stories and song request written in the radio station's Internet bulletin board. Our hypothesis is that when people request the same song, the situation or context in which they write the associated story is likely to be similar as well. Since each document contains a song request as well, by discovering similar documents, the system can recommend the songs linked to the similar documents.

There has been several approach in using contextual information as a criterion for recommending music. However, using textual stories written in the radio station bulletin board for music recommendation has not been attempted to the authors' knowledge. Another contribution would be that by implementing this approach to many existing SNS could provide a song that suits the message.

The remainder of the paper is organized as follows. In the next section, we first summarize recent music recommendation algorithms and address the problems of current approaches. We also introduce the characteristic of the stories written in the Korea radio station bulletin board. In Section III, we explain our system in detail. In Section IV, we provide a statistical evaluation of the system and in Section V, we present the results. We conclude the paper with a summary and directions for future work in Section VI.

## II. Background

There are several approaches in the music recommendation field. Commonly, the methods can be grouped into three categories depending on the algorithm: collaborative filtering method, content-based filtering method, and a hybrid method. Since our focus is on taking consideration of the user's input without any content analysis, we will discuss about some of the CF methods in this section.

### A. Collaborative Filtering-based Recommendation Systems

There are two different types of collaborative filtering method: the memory-based recommender system and the model-based recommender system. The memory-based recommender system again can be categorized into user-based CF and item-based CF depending on the focus of the algorithm. The user-based CF predicts the user's interest in a new item based on rating information from similar user profiles. The item-based CF works in a similar way but instead of using similar user profiles, it uses similarity between items [15].

On the other hand, the model-based recommender uses the collection of ratings to learn a model. Using the learned model, the expected rating for a new item is estimated. Some of the widely used models are the cluster model, Bayesian networks, statistical model, and machine learning models [1]. The performance of this method is greatly affected by the model and thus to create a model that improves the quality of the recommender system is still an ongoing issue.

While these CF methods are widely used in commercial nowadays, the effectiveness of the recommendation is questioned as the recommender systems confront some problems. Since the music database is extremely large, the rating information of the user is very sparse. This sparse user rating information can lead to biased suggestions. Another problem is that it neglects the contextual information of the listener. This problem has been tackled previously and will be explained in the following section.

### B. Context-aware Recommendation Systems

Reynolds *et al.* introduce the need to take into consideration of the user's contextual information. Through a survey experiment, they showed that activity, one of the contextual information, has a great impact on the listener's mood. From this research, it was shown that the activity one is involved in has great impact on the choice of the music one wants to listen to [11].

Su *et al.* proposed another method on using contextual information to recommend music [14]. Their system uses contextual information such as heartbeat, body temperature, air temperature, noise volume, humidity, light, motion, time, season, and location. Along with this contextual information, their system performs a content analysis on the music data to build a pattern database which links music with the user. Using this link and the contextual information, it proved to provide a more effective recommendation list.

However, the system suggested by Su *et al.* has an off-line preprocessing step which is to generate the pattern database via content analysis. Again, this confronts a scalability issue. A more critical problem is that while all the suggested contextual information might implicitly infer an activity state, it doesn't actually indicate what activity one is doing. In order to overcome the listed problems we suggest a recommendation system that uses the documents that the users themselves created. Within the document the user requests a song and describes the background for requesting the song. As mentioned above, since activity has great impact on the choice of the music one wants to hear, we believe that by using this document we could recommend song depending on the situation one is in.

### C. Characteristic of Korean Radio Broadcasting

What made our research possible was the characteristic of Korea radio broadcasting system. There are three participants in Korea radio channel: the DJ, celebrity guests, and the audience. The DJ and the celebrity guests direct their program and plays music that satisfy the theme for each section. The audience, mostly radio channel listeners, posts their personal stories along with a song request on the radio station bulletin board and these stories act as the pool of music to be selected by the radio DJ. Each story, which is written in Korean by a listener, is associated with a song request and a background for such request. We believe that the background information contains situational information. Thus, these documents, posted on the Korean radio channel's Internet bulletin board, provide a link between music and the contextual information. Therefore, we aim to use document similarity to find similar music.

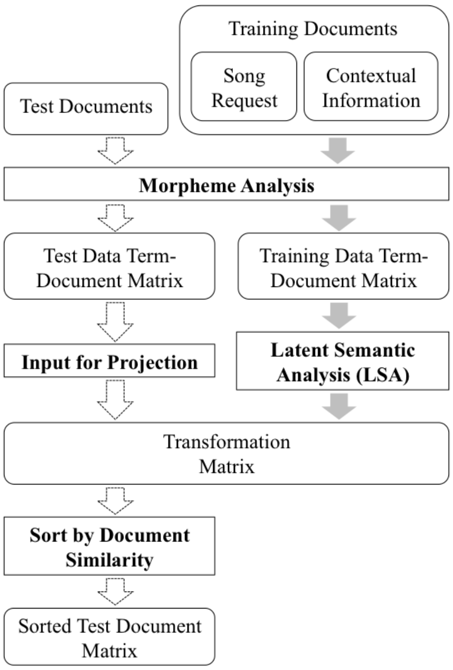


Figure 1. Overall process of our system. The evaluation process is shown altogether. The bold text indicates the steps of the process. The shaded arrow indicates the process of creating a transformation matrix and the unshaded arrow indicates the process of the evaluation.

## III. Proposed System

In this section, we describe our system that uses the textual data to extract the contextual information when recommending music. This approach can be expanded broadly as social network services are overwhelming these days and people tend to write about their situational status often. The overall system is shown in Figure 1. Amongst the stories gathered we divide them into test documents and training documents. Then, we perform a morpheme analysis on both data set. Latent Semantic Analysis (LSA) is performed on the training set to create a transformation matrix. Using this transformation matrix, the test documents are projected and ranked for evaluation. In the following sections we will talk about the core algorithm of our system in more detail.

### A. Morpheme Analysis

In our system, we use document similarity to generate a recommendation list. In order to find similarity between the documents, we first need a comparable representation of each story. This is accomplished by using the vector model. Each story is represented as a vector where each element represents the occurrence of the words used in the story. However, using all the words has two major problems. One problem is that the complete word set contains stop-words. Stop-words are words such as 'and', 'the', 'at', etc. These words are uninteresting words and the presence of them can degrade the performance of LSA. Another problem is that the complete word set contains stemmed words. For example, 'learn', 'learning', 'learned', and 'learnable' all come from the same stem 'learn' but is regarded as distinct words. This again causes the performance of LSA to go down. To avoid these problems, morpheme analysis is performed prior to vectorizing the documents [5]. The morpheme analysis tool removes the stop-words and also discovers the word stem.

### B. Latent Semantic Analysis

Through the morpheme analysis, each document is represented as a vector of word occurrence where the stop-words and stemmed words are removed. However, to use the vector as it is leads to another problem. Compared to the total word pool which is the overall words used in all of the gathered documents, the number of words used in each document is extremely small causing the vector to be sparse. Using this sparse vector for comparison would not be accurate. In the research field of language processing, Latent Semantic Analysis (LSA) has been used as a proper tool when comparing sparse data [6]. LSA processes the sparse matrix to discover the latent meaning of the documents or the words. With the processed word-document matrix containing latent meaning, we are now able to find similar documents by using distance metrics.

An important algorithm used when performing LSA is the singular value decomposition (SVD). After creating a word-document matrix by combining all the story vectors, the matrix is decomposed into a set of rotation and scale matrices. The result of the decomposition is shown in (1).

$$M = USV^T \tag{1}$$

where $M \in \mathbb{R}^{t\times d}$ is the original word-document matrix, $U \in \mathbb{R}^{t\times t}$ is the matrix representing the words, $S$ is a diagonal matrix of size $\mathbb{R}^{t\times d}$ containing singular values, and $V \in \mathbb{R}^{d\times d}$ is the matrix representing the documents. Both $U$ and $V$ are orthogonal.

Once the decomposition is done, the diagonal matrix $S$ and the document relevant orthogonal matrix $V^T$ are multiplied to find the semantic discriminations between the documents. The parameter that can be altered is the number of singular values to use. The number of singular values determines the dimension of the vector space where the reduced document vector will be projected to. The result of the projection can be shown as: $D' = S' \times V'^T$ where $D' \in \mathbb{R}^{k\times d}$ is the reduced approximation matrix, $S'$ is the reduced version of the diagonal matrix using $k$ singular

 ISBN: 978-1-61208-227-1 

values, and $V'$ is the reduced document relevant orthogonal matrix. After the projection, a distance metric is used to calculate the distance between the document vectors. In our experiments, we used cosine and Euclidean distance metrics.

### C. Music Recommendation based on Document Similarity

By performing LSA, a transformation matrix that would project the test document vector to the same vector space for comparison is created. We used 10,000 singular values for the reduction explained in Section III-B. Using the transformation matrix, the input matrix, which would be a set of test document vectors, is projected to the vector space and the distance between each vector is compared to generate a ranked list. The closer the vector is, the more similar the document will be. Therefore, assuming that people prefer similar music in similar situation, recommending music requested in similar documents would be a feasible recommendation.

## IV. Evaluation

In this section, we explain the dataset and the metrics used for evaluating our system. As mentioned in Section III-C, our assumption is that people in similar situation would prefer similar songs. Since our system extracts contextual information from individual stories, if our assumption is correct then the stories that request the same song would be similar. In order to validate our assumption, from the mined documents, we manually marked documents that requested the same song and regarded these as relevant to each other. We then applied conventional precision and recall approach and reciprocal rank to evaluate our system.

### A. Dataset

Individual stories are posted in the radio channel's Internet website. We data mined 14,000 documents from the bulletin board of the radio program held between 2:00 pm and 4:00 pm. Amongst the 14,000 documents, 10,000 documents were used to train the transformation matrix. The remaining 4,000 documents were used for evaluation. We first extracted the requested song for all 4,000 test documents. This was a semi-auto process since we had to manually mark the requested song for each document. After marking the data, we ran a program to check which music was requested how many times. Since the title and the musician can be a noise data, the program also deleted them after the counting process was performed. After counting the songs requested by the 4,000 test documents, we collected documents that were linked to the top 10 most frequently requested songs shown in Table (I). There were 291 documents altogether and these documents were used as a test set. From here on, documents requesting the same song will be denoted as relevant documents.

| Song ID | Song Title | Number of Documents |
|---|---|---|
| 1 | Happy Birthday to You | 41 |
| 2 | Heartbreaker | 40 |
| 3 | Can't I Love You | 36 |
| 4 | Relief | 37 |
| 5 | There Isn't Anyone Like You | 28 |
| 6 | Will you Marry Me | 24 |
| 7 | Cheer Up | 22 |
| 8 | I Don't Care | 22 |
| 9 | Tears are Bitter | 21 |
| 10 | Love Rain | 20 |

Table I
TOP 10 MOST FREQUENTLY REQUESTED SONGS.

### B. Metrics

Taking each document as an input, the remaining 290 documents were ranked based on document similarity. To measure similarity, we used Euclidean distance and cosine distance. We calculated three metrics to evaluate our system; mean average precision at 10 (MAP10), mean average precision at 5 (MAP5), and mean reciprocal rank (MRR). We compared the MAP5, MAP10 and MRR of our algorithm with that obtained when the documents were ranked randomly. From here on, the randomly generated MAP and MRR will be denoted as MAP5r, MAP10r and MRRr respectively.

*1) Mean Average Precision:* Precision and recall is an evaluation metric that is widely used in information retrieval. For each document, using equations (2) and (3), we calculated the precision until the recall rate reached 1. Then, it was averaged to find the average precision for each document. Once the average precision was calculated, we averaged the average precision for each test song.

$$Precision = \frac{relevant\_docs \cap retrieved\_docs}{retrieved\_docs} \tag{2}$$

$$Recall = \frac{relevant\_docs \cap retrieved\_docs}{relevant\_docs} \tag{3}$$

For each test song, we calculated the precision at 5 and 10. Since each song is associated with several relevant documents that requested the song, we calculated the precision at 5 and 10 for each input document and calculated the average of the mean precision for each input of the relevant document. These results were compared to the Mean Average Precision at 5 and 10 of that generated randomly. The random generation was assumed to have a uniform distribution and the result is show in Figure 2 and Figure 3.

*2) Mean Reciprocal Rank:* Another conventional metric in Information Science is the Mean Reciprocal Rank (MRR). In order calculate MRR, we first find the rank of the first relevant document for each document. After finding the first appearance of the relevant document for each test document, the average of the inverse of the rank is calculated. This

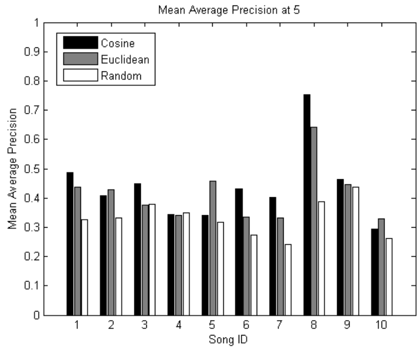


Figure 2. Mean Average Precision at 5 using cosine distance metric, euclidean distance metric, and randomly based metric.

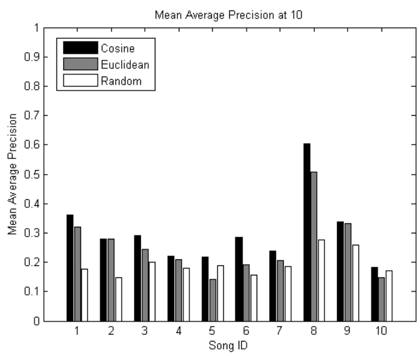


Figure 3. Mean Average Precision at 10 using cosine distance metric, euclidean distance metric, and randomly based metric.

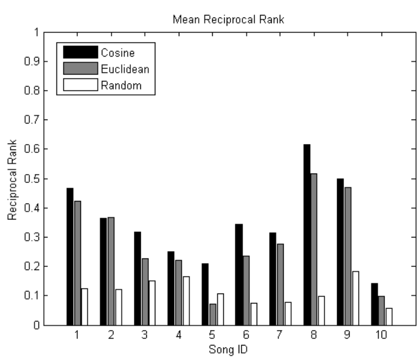


Figure 4. Mean Reciprocal Rank using cosine distance metric, euclidean distance metric, and randomly based metric.

result is again compared with that generated randomly and is shown in Figure 4.

## V. Results and Discussion

From the results shown in the previous section, we were able to find that cosine distance metric outperformed the euclidean distance metric. This can be explained by the fact that normalization wasn't performed prior to our evaluation. For example, a document using word 'a' once and 'b' once would be distinct from the document that uses word 'a' twice and word 'b' twice if Euclidean distance metric is applied. However, syntactically these two documents should be regarded as nearly the same. Cosine distance metric considers this fact and thus outperforms the Euclidean distance metric. Also, for most of the test data, the result of our algorithm outperformed the result generated randomly. This indicates that our system was able to find similar documents and those similar documents actually requested the same song. Thus, it is possible to recommend music purely based on textual mining and analysis of blog or specific music related programs. The best performance was shown in song 9. A possible explanation is that the lyric and the melody of the song matches. Song 9 is a ballad song which is quiet and moody. The lyric is about reminiscence of one's past love. The lyric follows the moody melody and thus people who request this song would share a similar situation regarding sad love.

However, as shown in song 10, there were cases where our algorithm didn't perform well. This can be explained by the characteristic of the song. The melody of the song is bright and cheerful. However, the lyric of the song is about waiting for love. Having such characteristic, people whose preference is more dependent on the melody might prefer the song in a cheerful situation while people whose preference is more dependent on the lyric might prefer the song in an moody situation reminiscing love. We believe that such diversity in situation when requesting the song is the reason for the poor result.

An unexpected finding was the relatively poor result for song 6. The lyric and the melody of the song absolutely fits for proposing marriage. Therefore, our expectation was that the music would be requested usually in situations regarding marriage proposal. However, when the documents requesting this song were checked manually, we found out that the song was requested not only in proposing situations, but also in situations when the user was celebrating his/her anniversary. Due to this subtle difference, our system was not able to find similarity between marriage and anniversary, and thus gave a relatively low result. In order to overcome these situations, future work will be discussed in the following section.

Despite some limitations, most of the results outperformed the results obtained when the stories were randomly ranked. Thus, our assumption that people shared similar preference in similar situation proved right and our approach to analyze textual information to gather contextual information showed possibilities.

## VI. Conclusion

In this paper, we presented a novel approach to recommending music based on text analysis. Rather than implicitly guessing the contextual information of the users, we showed that it is possible to use documents, written by the users, to extract contextual information explicitly. To this end, we gathered radio stories written by individuals via the radio channel's bulletin board and performed LSA to identify the semantic meanings of the documents to find similar stories. Our assumption was that if people shared similar situational information, then the music they prefer would be similar. Since each story was associated with a song request, the song linked to the most similar story could be recommended. In order to evaluate the system, we used several metrics to check if similar stories actually requested the same song. The result showed that there was a positive correlation between story similarity and song similarity, and thus it could be possible to recommend music purely based on document analysis. Additionally, to check the quality of the recommendation, we plan to perform an user evaluation test.

One limitation in our experiment was that the documents retrieved were limited to stories written in the Korean radio station's bulletin board. However, the main purpose of this research was to check the validity of the approach in using text mining and analysis to extract the contextual information of the user when recommending music. While not perfect, we showed that in most cases, people requested similar songs in similar situation. Thus, it proved the possibility of performing text analysis when recommending music. To expand our research for worldwide radio listeners remains as a future work.

Along with applying our research to worldwide listeners, we plan to improve the quality of the system. As indicated above, LSA has some limitations. One most crucial limitation is that it is not appropriate for detecting polysemies. Polysemies are words that have multiple meanings. Recent studies have shown that this problem can be tackled by implementing the probabilistic Latent Semantic Analysis (pLSA). Therefore, by performing pLSA to our dataset we expect to achieve a better result.

Another possible improvement can be found in the morpheme analysis tool. The morpheme analysis tool we used, correctly removed stop-words almost completely but was only able to discover the stem words with approximately 80% correctness. This rate went down significantly if the document contained misspelled words, abbreviations, and non-spaced words. We expect that our approach will have better performance if these noise within the stories are handled.

## VII. Acknowledgements

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0013476).

## References

[1] G. Adomavicius and A. Tuzhilin, *Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions*, IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 6, pp. 734–749, 2005.

[2] C. Anderson, *The long tail*, Gramedia Pustaka Utama, 2006.

[3] Y. Hu, Y. Koren, and Volinsky, *Collaborative filtering for implicit feedback datasets*, Proc. of ICDM-08, 8th IEEE ICDM, pp. 263–272, 2008.

[4] International Federation of the Phonographic Industry, *Digital Music Report 2012*, Available at: http://www.ifpi.org/content/library/DMR2012.pdf (Accessed: 10 September 2012).

[5] Kookmin University Korean Language Technology, Available at: http://nlp.kookmin.ac.kr/HAM/kor/index.html (Accessed: 13 October 2012).

[6] T.K. Landauer, P.W. Foltz, and D. Laham, *An introduction to Latent Semantic Analysis*, Discourse Processes, Vol. 25, No. 2–3, pp. 259–284, 1998.

[7] T. Li and M. Ogihara, *Content-based music similarity search and emotion detection*, Proc. IEEE Int. Conf. Acoustic, Speech, Signal Processing, Vol. 5, pp. 705–708, 2004.

[8] Q. Li, B.-M. Kim, D.-H. Guan, and D.-W. Oh, *A music recommender based on audio features*, SIGIR 04: Proc. 27th Annual Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, pp. 532–533, 2004.

[9] S. Pauws, W. Verhaegh, and M. Vossen, *Fast generation of optimal music playlists using local search*, Proc. 7th Int. Conf. Music Inf. Retrieval, pp. 138–143, 2006.

[10] R. Ragno, C. J. C. Burges, and C. Herley, *Inferring similarity between music objects with application to playlist generation*, Proc. 7th ACM SIGMM Int. Workshop Multimedia Inf. Retrieval, pp. 73–80, 2005.

[11] G. Reynolds, D. Barry, T. Burke, and E. Coyle, *Interacting with large music collections: Towards the use of environmental metadata*, Multimedia and Expo, 2008 IEEE International Conference, pp. 989–992, 2008.

[12] A. I. Schein, A. Popescul, L. H. Ungar, and D. M. Pennock, *Methods and metrics for cold-start recommendations*, Proc. of the 25th SIGIR, pp. 253–260, ACM Press, 2002.

[13] B. Schwartz, *The paradox of choice: Why more is less*, Harper Perennial, 2005.

[14] J.H. Su, H.H. Yeh, P.S. Yu, and V.S. Tseng, *Music recommendation using content and context information mining*, Intelligent Systems, IEEE, Vol. 25, No. 1, pp. 16–26, 2010.

[15] J. Wang, A. P. de Vries, and M. J. T. Reinders, *Using user-based and item-based collaborative filtering approaches by similarity fusion*, Proc. of the 29th SIGIR, 2006.

[16] K. Yoshii, M. Goto, K. Komatani, T. Ogata, and H. G. Okuno, *Hybrid collaborative and content-based music recommendation using probabilistic model with latent user preferences*, Proc. ISMIR, 2006.

[17] M. Zhang and N. Hurley, *Avoiding monotony: Improving the diversity of recommendation lists*, Proc. of the ACM RecSys, pp. 723-732, ACM, 2008.

 ISBN: 978-1-61208-227-1 