An MPD Player with Expert Knowledge-based Single User Music Recommendation

Hen-Hsen Huang
hhhuang@nccu.edu.tw
Department of Computer Science, Natinal Chengchi University
MOST Joint Research Center for AI Technology and All Vista Healthcare
Taipei City, Taiwan

ABSTRACT

This work demonstrates a music player based on Music Player Daemon (MPD), a protocol popular for audiophiles, with personalized music recommendation. As a standalone player, we focus on the scenario of single user recommendation. Different from collaborative filtering based recommendation, which relies on usage patterns from a large number of users, we propose a novel approach that does not require other users' information. We formulate the recommendation as a task of knowledge base completion and exploit the expert knowledge from a music knowledge base. The effectiveness of our approach is evaluated, and the player is released as an open-source software for music lovers.

CCS CONCEPTS

• **Information systems → Recommender systems**; *Personalization*; • **Computing methodologies** → *Knowledge representation and reasoning*.

KEYWORDS

music recommendation, personalization, knowledge base completion, expert knowledge representation

ACM Reference Format:

Hen-Hsen Huang. 2019. An MPD Player with Expert Knowledge-based Single User Music Recommendation. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI '19 Companion), October 14–17, 2019, Thessaloniki, Greece.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3358695.3360919

1 INTRODUCTION

Modern approaches to music recommendation are mainly developed for online content distribution platforms such as Spotify.[1] For example, collaborative filtering-based approach extracts the association between customers and items (i.e. albums or tracks) from a large amount of logs from numerous users [4, 6, 8, 10]. On the other hand, content-based approach models the association between customers and the acoustic features extracted from a wide range of audio contents [12]. Logs from numerous users are only accessible to content distributors, and the full coverage of audio contents are possessed by content providers. In other words, both kinds of information are exotic resources exclusive to enterprises. The preference of music being manipulated by a few giant companies can be a serious issue for both the progress of the music industry and the freedom of customers: Choices are limited by the commercial cooperation, and the diversity is clamped down.

Some music lovers keep away from online music services and lend toward maintain their own recording collections. They purchase hi-definition audio files from the Internet or rip the tracks from CDs/SACDs and store the audio files in their own computer or network attached storage (NAS). For music listening, they play an album from their collections with standalone music players, and no giant company is involved in monitoring and controlling their behaviors. For these users, however, suggestions about the next album to purchase or the next artist to explore are still somewhat desired. Thus, we present a standalone music player that is equipped with a music recommendation system based on an alternative approach that does not require others' logs and audio contents.

This work investigates the music recommendation from a novel perspective. Instead of collaborative filtering-based and content-based approaches, our model predicts the next album to purchase or the next artist to explore by knowledge base inference. Previous work also explored knowledge base approaches to music recommendation [13], a path-based approach was proposed to predict the similarity between a user and a song. However, they did not focus on the scenario of single user recommendation. Instead, large amount of user logs were used to train the path ranking model for recommendation. In their knowledge base,[2] only a few relations such as *singer of a song*, *composer of a song*, *lyricist of a song*, and *genre* are defined. No social relations among artists are considered.

Different from previous work, we formulate the recommendation as a special case of knowledge base completion. Due to the lack of the information from other users, our methodology heavily relies on the expert knowledge. A rich music knowledge base is constructed by extracting expert knowledge from a long-established online music database, All Music Guide.[3] All musicians and recordings are entities in the knowledge base, and their relations are labeled in several types. Our knowledge base contains not only the fundamental information such as styles, genres, themes, and moods of musicians and recordings, but also the social network of musicians. The relationships among musicians such as similar,

[1] www.spotify.com

[2] www.kaggle.com/c/kkbox-music-recommendation-challenge

[3] www.allmusic.com

WI '19 Companion, October 14–17, 2019, Thessaloniki, Greece

ACM ISBN 978-1-4503-6988-6/19/10…$15.00
https://doi.org/10.1145/3358695.3360919

influenced by, and followed by are taken as important information for modeling the knowledge base inference.

Our recommendation model is integrated into a cross-platform, open-source music player, which is based on the Music Player Daemon (MPD)[4] protocol and can be easily used as a client for playing one's music collection. Based on the playback logs collected by the player, a personalized music knowledge base is built for recommending the albums to purchase or the artists to explore for the single user.

The contributions of this work are threefold as follows.

(1) We present an alternative approach to music recommendation. By integrating expert knowledge, experimental results show the effectiveness of single user music recommendation.
(2) The approaches based on collaborative filtering suffer from the issue of bandwagon effects [9]. In contrast, our model for personalized music recommendation respects the user's personal preferences more.
(3) We release our MPD player with the music recommendation model as an open-source software for the research community and the music lovers who object to be controlled by giant companies.[5]

The rest of this paper is organized as follows. The Section 2 shows the rich knowledge base built for music recommendation. Our model for recommendation, adapted from the task of knowledge base completion, is described in Section 3. A brief evaluation of our system is shown in Section 4. Section 5 describes the features of our music player, and Section 6 concludes this work.

2 MUSIC KNOWLEDGE BASE

The music knowledge base plays the crucial role in our system. We construct a music knowledge base with the data crawled from All Music Guide, a large online music database comprised of the information of three millions of albums with metadata such as the styles, themes, genres, and ratings well-annotated by experts since 1991. One important information available from the database is the social network of artists. The social relations among artists are labeled in six categories, including SIMILARS, INFLUENCERS, FOLLOWERS, GROUPMEMBERS, ASSOCIATEDWITH, and COLLABORATORWITH. Specifically, an artist is either a musician or a group. For example, the Beatles is an artist, John Lennon is also an artist, and their relations are denoted as GROUPMEMBER(The Beatles, John Lennon). Table 1 shows the entities in the current version of our music knowledge base, which covers a total of 91,929 entities and 527,121 factual triples. All types of relations between a pair of entities in the knowledge base, including the six social relationships, are summarized in Table 2. Compared to the knowledge base used in previous work [13], our music knowledge base provides much richer information.

3 RECOMMENDATION MODEL

Knowledge base completion is aimed at predicting the missing relation between two entities. For example, a musician's membership of a rock band is not labeled in a knowledge base, and the model for knowledge base completion is expected to inference the fact $E_h + E_r \approx E_t$, where E_h is the embedding of the head entity h (i.e. the rock band), E_t is the embedding of the tail entity t (i.e. the musician), and E_r is the embedding of their relation r, GROUPMEMBERS.

Approaches to knowledge base completion are roughly categorized into the embedding-based method like TransE [3] and the path-based method like path ranking [7]. In this work, we formulate music recommendation as a task of knowledge base completion by considering the user as a special entity in the knowledge base with relations to preferred artists/albums. As illustrated in Figure 1, we add the user as an entity into the knowledge base and also add the LIKE relations between the user and those albums most played, since the log of each track has been played is captured by the MPD player.

Then, we perform the knowledge embedding algorithm to model the relations among a variety of entities. In other words, we treat the user behaviour as facts and train the knowledge embedding model to inference user's preference. As a result, the distance between the user u and an album or an artist t can be estimated as follows.

$$\delta(u, t) = ||E_u + E_{\text{LIKE}} - E_t||_1 \quad (1)$$

where $|| \cdot ||_1$ is the L_1 norm. Then, the recommendation of next albums to purchase for the user u can be made by selecting those albums with a distance lower than a threshold τ as follows.

$$\hat{A} = \{t | t \in A \wedge t \notin A_u \wedge \delta(u, t) < \tau\} \quad (2)$$

where A is the collection of all albums in the knowledge base, A_u are the albums already in the user's personal collection, τ is a hyperparameter denoting the threshold of the distance, and $\hat{A}$ is the set of recommended albums. For the case in Figure 1, Pink Floyd's album *The Wall* will be recommended since the user enjoys the albums of the same and the similar artists. The recommendation of next artists to explore can be made by a similar fashion.

Type	Instances	Samples
Artist	11,162	The Beatles, Bob Dylan, U2
Album	79,488	Revolver, The Wall, Achtung Baby
Theme	187	Introspection, Reflection, Late Night
Mood	295	Playful, Reflective, Earnest
Style	776	Alternative, Indie Rock
Genre	21	Pop/Rock, R&B, Electronic, Jazz

Table 1: Statistics of our music knowledge base

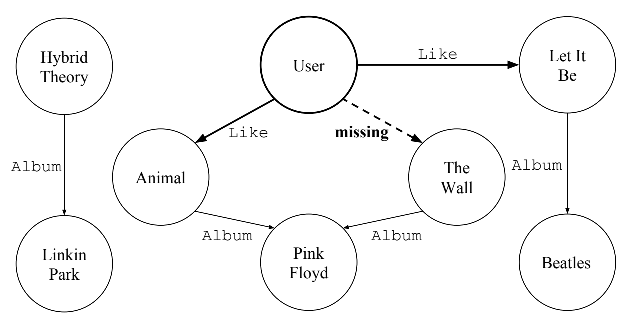

Figure 1: Recommendation made by recovering the missing link between the user and an album

[4]https://www.musicpd.org
[5]https://github.com/hhhuang/mpd_player

	Relation	Type of Entity 1	Type of Entity 2	Description
Social relationship	Similars	Artist	Artist	Similar artists
	Influencers	Artist	Artist	Artists who influenced the artist
	Followers	Artist	Artist	Artists who followed the artist
	Group Members	Artist	Artist	Artists who are the group member of the artist
	Associated With	Artist	Artist	Artists who are associated with the artist
	Collaborator With	Artist	Artist	Artists who have ever collaborated with the artist
Property	Album	Artist	Album	The albums of the artist
	Theme	{Artist, Album}	Theme	Themes usually cared about by the artist
	Mood	{Artist, Album}	Mood	Moods usually expressed in the artist's albums
	Style	{Artist, Album}	Style	Styles of the artist
	Genre	{Artist, Album}	Genre	Genre of the artist

Table 2: Relations of our music knowledge base

Task	k	Accuracy	MRR
Next Artist	4	0.667	0.799
Next Artist	10	0.500	0.638
Next Artist	50	0.167	0.290
Next Album	4	0.619	0.802
Next Album	10	0.476	0.666
Next Album	50	0.333	0.475

Table 3: Performance of link prediction for next artist to explore and next album to purchase

4 EVALUATION

We split an eight-year user log into two parts chronologically. The log records from the first five years are used for training the knowledge base, and the log records from the last three years are reserved for testing. We train the knowledge embeddings with the OpenKE package [5], which consists of a number implementations of famous algorithms for knowledge base representation. The TransE [3] model is employed in our system. The dimension of knowledge embeddings is 50, and the batch size is 100. An equal number of instances are selected as the negative samples for the entities. According to our scoring function for measuring the distance between the user and an album/artist, the loss function is the L_1 norm.

We evaluate our model with the task of link prediction at $k \in \{4, 10, 50\}$, where k is the number of candidates. For example, the model has to rank the positive instance (i.e. an album was frequently played from the last three years) higher than other three negative instances for the evaluation at $k = 4$. Table 3 reports the performance of our recommendation model in Accuracy and MRR (mean reciprocal rank). Experimental results show that the performance remains robust even though a number of negative instances are added to mix up. We also explore more recent knowledge embedding models such as Bilinear-Diag[14] and Complex Embeddings [11], however, the simple TransE model outperforms the advanced ones in our application.

5 THE MUSIC PLAYER

Our MPD player with the recommendation system was implemented in Python 3. It is an open-source project for Windows, Mac, and Linux. The player automatically records the user events including an album being added into the collection and a track being played. The resultant logs are then analyzed by the player for music recommendation.

As shown in Figure 2, the albums most likely to purchase are listed in an additional tab in our player. The album covers are automatically fetched from the Internet for a better user experience. In addition to single user music recommendation, the key features of our music player are listed as follows.

- The player is friendly with a large number of albums. It has been tested with a collection consisting of 2,000 CDs.
- This player is album-oriented. All the tracks in the MPD server are reorganized into albums. Compared with individual tracks, album is a much more structural and meaningful unit for serious music lovers' critical listening. Definitely, the user can also play a single track from an album.
- Many MPD servers and renderers are built on low-power devices such as NAS and Raspberry Pi. [6] To reduce the load of these devices, most of the load is taken by the controller side (i.e. this player). This design philosophy results a slim, minimal, somewhat slow player, but both server/renderer sides benefit from stability, lower jitter, and better sound quality [1, 2]. This property is very important to audiophiles.

6 CONCLUSION

Modern approaches to music recommendation are mostly exclusive to the online music service platforms that possess countless user logs or a wide range of audio contents. This paper demonstrates an alternative approach to personalized music recommendation for the music lovers who keep away from the giant companies. Our recommendation system is smoothly integrated into an audiophile player based on the MPD protocol. Without the need of numerous users' logs, our novel approach exploits the single user's log with expert knowledge.

To the best of our knowledge, this is the first attempt to formulate music recommendation as a task of knowledge base completion. A comprehensive evaluation and user study will be conducted in the future work. This work considers the structured metadata labeled on the All Music Guide website. Unstructured textual information

[6] www.raspberrypi.org

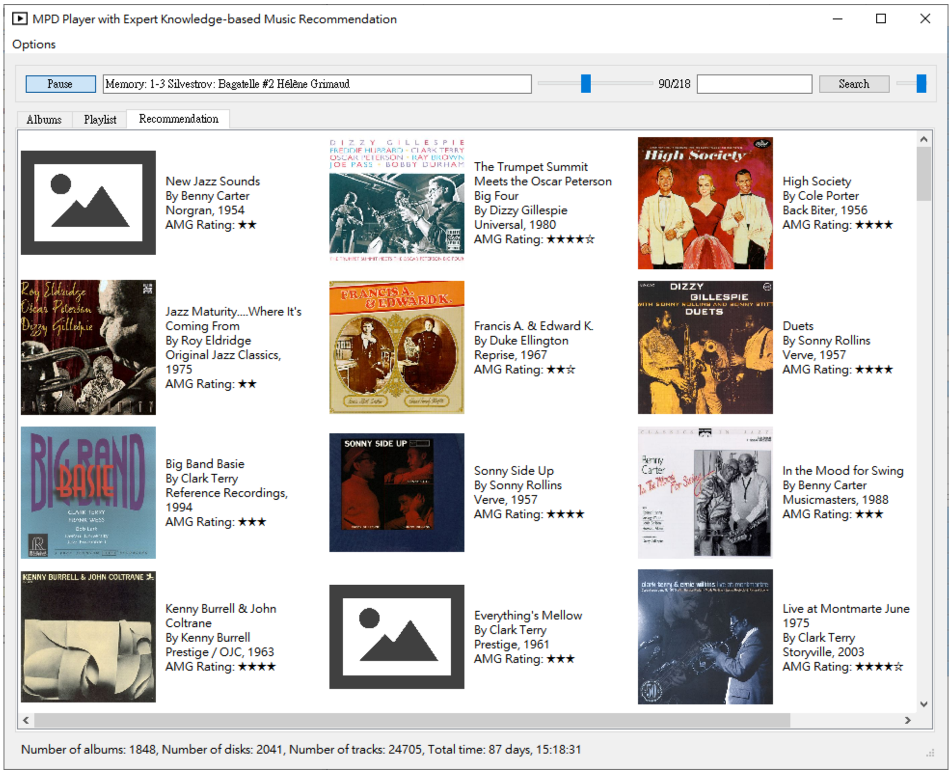

Figure 2: Personalized album recommendation

such as album reviews and artist biographies can also be integrated into our recommendation model to exploit the other kind of expert knowledge.

ACKNOWLEDGMENTS

This research was partially supported by the Ministry of Science and Technology, Taiwan, under the grant MOST-108-2634-F-002-008-.

REFERENCES

[1] Kaoru Ashihara, Shogo Kiryu, Nobuo Koizumi, Akira Nishimura, Juro Ohga, Masaki Sawaguchi, and Shokichiro Yoshikawa. 2005. Detection threshold for distortions due to jitter on digital audio. *Acoustical science and technology* 26, 1 (2005), 50–54.

[2] Eric Benjamin and Benjamin Gannon. 1998. Theoretical and Audible effects of jitter on Digital Audio Quality. In *Audio Engineering Society Convention 105*. Audio Engineering Society.

[3] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (Eds.). Curran Associates, Inc., 2787–2795. http://papers.nips.cc/paper/5071-translating-embeddings-for-modeling-multi-relational-data.pdf

[4] Chih-Ming Chen, Chun-Yao Yang, Chih-Chun Hsia, Yian Chen, and Ming-Feng Tsai. 2016. Music Playlist Recommendation via Preference Embedding. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys '16)*. ACM, New York, NY, USA, 2.

[5] Xu Han, Shulin Cao, Xin Lv, Yankai Lin, Zhiyuan Liu, Maosong Sun, and Juanzi Li. 2018. OpenKE: An Open Toolkit for Knowledge Embedding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, Brussels, Belgium, 139–144. https://www.aclweb.org/anthology/D18-2024

[6] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural Collaborative Filtering. In *Proceedings of the 26th International Conference on World Wide Web (WWW '17)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 173–182. https://doi.org/10.1145/3038912.3052569

[7] Bhushan Kotnis, Pradeep Bansal, and Partha P. Talukdar. 2015. Knowledge Base Inference using Bridging Entities. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, 2038–2043. https://doi.org/10.18653/v1/D15-1241

[8] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based Collaborative Filtering Recommendation Algorithms. In *Proceedings of the 10th International Conference on World Wide Web (WWW '01)*. ACM, New York, NY, USA, 285–295. https://doi.org/10.1145/371920.372071

[9] S. Shyam Sundar, Anne Oeldorf-Hirsch, and Qian Xu. 2008. The Bandwagon Effect of Collaborative Filtering Technology. In *CHI '08 Extended Abstracts on Human Factors in Computing Systems (CHI EA '08)*. ACM, New York, NY, USA, 3453–3458. https://doi.org/10.1145/1358628.1358873

[10] Thanh Tran, Renee Sweeney, and Kyumin Lee. 2019. Adversarial Mahalanobis Distance-based Attentive Song Recommender for Automatic Playlist Continuation. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*. ACM, New York, NY, USA, 245–254. https://doi.org/10.1145/3331184.3331234

[11] Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex Embeddings for Simple Link Prediction. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48 (ICML'16)*. JMLR.org, 2071–2080. http://dl.acm.org/citation.cfm?id=3045390.3045609

[12] Aaron van den Oord, Sander Dieleman, and Benjamin Schrauwen. 2013. Deep content-based music recommendation. In *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (Eds.). Curran Associates, Inc., 2643–2651. http://papers.nips.cc/paper/5004-deep-content-based-music-recommendation.pdf

[13] Xiang Wang, Dingxian Wang, Canran Xu, Xiangnan He, Yixin Cao, and Tat-Seng Chua. 2018. Explainable Reasoning over Knowledge Graphs for Recommendation. *CoRR* abs/1811.04540 (2018). arXiv:1811.04540 http://arxiv.org/abs/1811.04540

[14] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2014. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575* (2014).