MusicRoBot: Towards Conversational Context-Aware Music Recommender System

Chunyi Zhou[1], Yuanyuan Jin[1], Kai Zhang[2], Jiahao Yuan[1], Shengyuan Li[1], and Xiaoling Wang[1](✉)

[1] Shanghai Key Laboratory of Trustworthy Computing, MOE International Joint Lab of Trustworthy Software, East China Normal University, Shanghai, China
cyzhou@stu.ecnu.edu.cn, xlwang@sei.ecnu.edu.cn

[2] Shenzhen Gowild Robotics Co. Ltd., Shenzhen, China
zhangkai_ai@gowild.cn

Abstract. Traditional recommendation approaches work well on depicting users' long-term music preference. However, in the conversational applications, it is unable to capture users' real time music taste, which are dynamic and depend on user context including users' emotion, current activities or sites. To meet users' real time music preferences, we have developed a conversational music recommender system based on music knowledge graph, *MusicRoBot* (Music RecOmmendation Bot). We embed the music recommendation into a chatbot, integrating both the advantages of dialogue system and recommender system. In our system, conversational interaction helps capture more real-time and richer requirements. Users can receive real time recommendation and give feedbacks by conversation. Besides, *MusicRoBot* also provides the music Q&A function to answer several types of musical question by the music knowledge graph. A WeChat based service has been deployed piloted for volunteers already.

Keywords: Music recommendation · Online recommendation
Dialogue system · Recommender system

1 Introduction

Listening to music has been common during many people's leisure time. Generally, the recommended content includes hit songs, daily playlist and music radio, but these kind of interactive ways limit the expression of requirements. In this work, we have implemented a conversational music recommender system, *MusicRoBot* (Music RecOmmendation Bot), which embeds music recommendation

X. Wang—This work was supported by NSFC grants (No. 61472141), Shanghai Knowledge Service Platform Project (No. ZF1213) SHEITC and Shanghai Agriculture Applied Technology Development Program (No. G20160201).

J. Pei et al. (Eds.): DASFAA 2018, LNCS 10828, pp. 817–820, 2018.
https://doi.org/10.1007/978-3-319-91458-9_55

into a ordinary chatbot innovatively. Comparing with traditional recommendation scenarios, there're many differences in the conversational scenario: (1) it emphasizes more on online interactions; (2) this scenario is more context-sensitive; (3) dialogues carry richer but more complex information. Obviously, conversational recommendation is significant but challenging. In fact, conversational recommendation has been already studied. Christakopoulou et al. [1] proposes a conversational recommender system for restaurant recommendation by asking user absolute or relative questions. Sun et al. [2] demonstrates a conversational products recommendation agent based on deep learning technologies, but this demo seems like a virtual sales agent using a task-oriented dialog system. Besides, different from most existing chatbot, we focus on music-domain rather than open-domain, and we also construct *Music Knowledge Graph* (MKG) in support of musical entity recognition and recommendation.

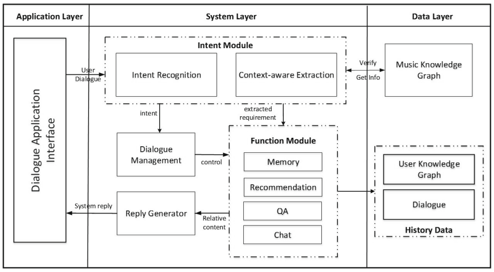

Fig. 1. Architecture of MusicRoBot

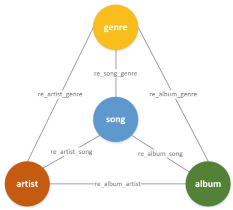

Fig. 2. Entities and relationships in *MKG*

2 System Design

Our system can be is divided into three layers: data layer, system layer and application layer. Figure 1 shows the architecture of *MusicRoBot*.

2.1 Music Knowledge Graph (MKG)

In support to better recognition and recommendation, we construct *Music Knowledge Graph*, raw data of which comes from *Xiami*[1]. We organize all musical entities as a graph in consideration of advantages in inference and analysis. Figure 2 shows entities and their relationships, the entity is represented as node and the relationship as edge. We define four types of entities: song, album, artist and genre, and our genres include both professional genres and common tags. *MKG* is stored in neo4j[2], and there're currently over 6 million songs, 600 thousand albums, 130 thousand artists, nearly 500 genres and still increasing.

[1] xiami's homepage: http://www.xiami.com/.
[2] neo4j's homepage: https://neo4j.com/.

2.2 Scenario Design

In our demo, we have designed four scenarios as follow:

Memory-based User Portrait Construction. In Fig. 1, it's short for *Memory*. System can capture users' basic properties and preferences on music during dialogues and store in *User Knowledge Graph (UKG)*. These properties can be used as explanation for recommendations and relieve the cold-start problem. In this implementation, basic property contains age, gender and current emotion. Preference contains all kinds of music entities in *MKG*. In addition, we conduct collision detection on user's basic properties.

Q&A. *Q&A* consists of user properties' Q&A and music knowledge's Q&A. This function is mainly designed for enhancing the interactivity between user and system. It may help discover useful user preferences for our future research.

Recommendation. It's the core module in this system. There're three kinds of recommendation scenarios: specific-query based recommendation, free recommendation and emotion-based recommendation. The recommended items include song, album and artist. Besides, in coping with online and interactive recommendation, we adopt a bandit-based recommendation algorithm, C^2UCB [3]. We compare C^2UCB with most popular strategy using *Xiami* user's listened song list and show the average reward (AR) for each user in Table 1. The result shows that user prefer less popular songs and prove the advantage of C^2UCB.

Chat. This function is in charge of the other scenarios which don't match the above situations. It is essential but not our focus, we employ the existed implementation by *emotibot*[3].

2.3 Intent Recognition and Dialogue Management

In ***Intent Module***, system recognizes user intent and extracts useful constraints from input. We summarize this task into *Intent Recognition* and *Realtime Requirements Extraction*. This module is implemented by ***Gowild***[4], applying both template matching and classifier. ***Dialogue Management*** is in charge of making next system action, which plays a role as a central controller. Both users' current intents and the context of previous dialog are considered during decision making.

3 Demonstration

Our demo is published as a *WeChat Service* and it currently supports only Chinese text input. Figure 3 shows the representative example motion-based recommendation scenario: a new user comes in and expresses his negative emotion, in this case, system inquires user preferences under this emotion, then recommends song as normal. Currently, system provides multi-turn recommendations at most three times, when user doesn't accept recommendations.

[3] *emotibot*'s homepage: http://www.emotibot.com.
[4] *Gowild*'s homepage: http://www.gowild.cn.

Table 1. Comparison with Most-popular recommendation

#round	#user	AR for MP	AR for C^2UCB based	Promotion (%)
1	645	0.40	0.41	2.5
10	645	4.09	4.5	10.02
20	645	8.51	9.24	8.58
50	645	20.09	23.87	18.82
100	645	39.84	48.27	21.16
200	504	80.21	98.87	23.26
500	277	200.46	261.77	30.58

Fig. 3. WeChat Service Demonstration for the recommendation scenario

References

1. Christakopoulou, K., Radlinski, F., Hofmann, K.: Towards conversational recommender systems. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 815–824. ACM (2016)
2. Sun, Y., Zhang, Y., Chen, Y., et al.: Conversational recommendation system with unsupervised learning. In: Proceedings of the 10th ACM Conference on Recommender Systems, pp. 397–398. ACM (2016)
3. Qin, L., Chen, S., Zhu, X.: Contextual combinatorial bandit and its application on diversified online recommendation. In: Proceedings of the 2014 SIAM International Conference on Data Mining, pp. 461–469. Society for Industrial and Applied Mathematics (2014)