Music Playlist Recommendation via Preference Embedding

Chih-Ming Chen, Chun-Yao Yang, Chih-Chun Hsia, Ming-Feng Tsai

Department of Computer Science, National Chengchi University

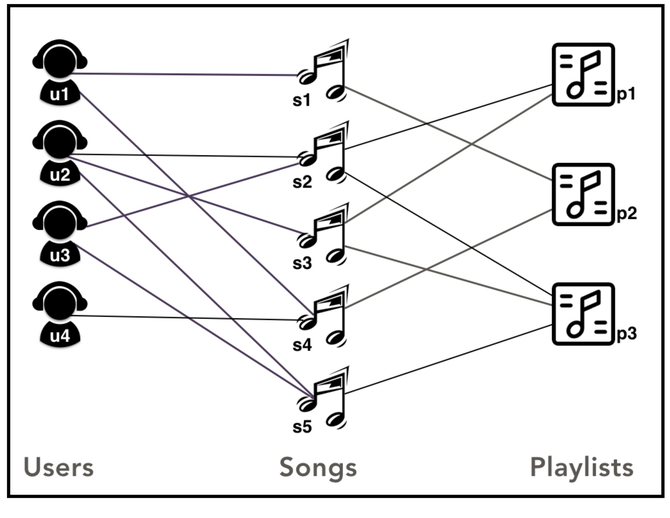
Yian Chen

Research Center, KKBOX Inc.

Introduction

In the music streaming services, such as Spotify and KKBOX, there are lots of playlists created by users and service providers. In this work, we aim to recommend such playlists to the users. Specifically, we propose a preference embedding method based on a user-song-playlist graph to learn the preference representations of these three types of entities.

Figure 1



Methodology

Step1. Graph Construction (Figure 1)

- Each entity (a user, a song, or a playlist) is a vertex.
- Each vertex is connected by two types of edges:
  1. user-song: the user preference over songs.
  2. playlist-song: the relations between the songs and the playlist.

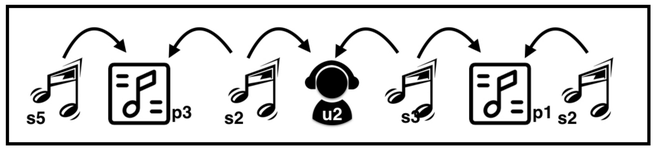
Step2. Entity Embedding Learning (Figure 2)

- Iteratively updates each vertex representation, Φ, according to its proximity to the sampled vertex.
- Minimize the set of sampled target-to-proximity pairs.

Step3. Distance Calculation

- Calculate the Euclidean distance between embeddings of the target user and playlists.
- Recommend the nearest *N* playlists to the target user.

Figure 2



$$O_v = \begin{cases} -\sum_{s \in Pref(u)} \log p(s|\Phi(u)) & \text{if } v \in U \\ -\sum_{s \in Plist(p)} \log p(s|\Phi(p)) & \text{if } v \in P \end{cases}$$

Experiments

Dataset: KKBOX

#User	#Song	#Playlist
50,000	400,000	130,000

1. Data splitting: for every user, 50%-50% for training/testing.
2. Ground truth: If a playlist includes over 70% user preference, We consider the user likes this playlist.

Performance:

	Pop	DW, w=2	DW, w=6	LINE, 2nd	HPE, w=2	HPE, w=6
P@5	1.88%	12.85%	**13.19%**	7.14%	10.39%	12.22%
P@10	3.67%	12.16%	12.37%	6.66%	10.29%	**12.52%**
P@15	5.39%	11.69%	11.88%	6.30%	10.12%	**12.36%**
P@20	7.08%	11.32%	11.56%	6.05%	9.99%	**12.35%**

DW: DeepWalk, w: window size,
HPE: Heterogeneous Preference Embedding

Future Work

1. Add extra information into the network to robust the heterogeneous graph.
2. Carry out more advanced context-aware recommendations.

RecSys 2016