Leveraging the structure of musical preference in content-aware music recommendation

Paul Magron, Cédric Févotte

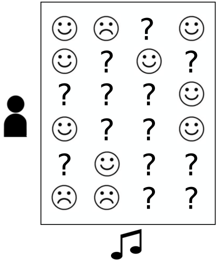
CNRS, IRIT, Université de Toulouse, France

ICASSP2021

Introduction

Music recommendation

- Predict items (=songs) that a user might be interested in.
- A task at the core of many commercial platforms (Spotify, Deezer...).
- Main idea: exploiting *similarities* between users and/or items.



Challenges [Schedl, 2018]

- Playlist continuation (sequential recommendation).
- Context-aware (situation, personality, demography).
- **Cold-start recommendation.**

Schedl et al., **Current challenges and visions in music recommender systems research**, *IJMIR*, 2018.

Recommendation approaches

Collaborative filtering: Users with similar tastes in the past (collected data) will have similar tastes in the future (predictions).

	Performance	Cold-start
Collaborative filtering	✓	✗

Recommendation approaches

Collaborative filtering: Users with similar tastes in the past (collected data) will have similar tastes in the future (predictions).

Content-based approaches: Users who liked some songs (collected data) will like songs with a similar content (predictions).

	Performance	Cold-start
Collaborative filtering	✓	✗
Content-based	✗	✓

Recommendation approaches

Collaborative filtering: Users with similar tastes in the past (collected data) will have similar tastes in the future (predictions).

Content-based approaches: Users who liked some songs (collected data) will like songs with a similar content (predictions).

Content-aware recommendation: incorporate content as side information in collaborative filtering.

	Performance	Cold-start
Collaborative filtering	✓	✗
Content-based	✗	✓
Content-aware filtering	✓	✓

Recommendation approaches

Collaborative filtering: Users with similar tastes in the past (collected data) will have similar tastes in the future (predictions).

Content-based approaches: Users who liked some songs (collected data) will like songs with a similar content (predictions).

Content-aware recommendation: incorporate content as side information in collaborative filtering.

	Performance	Cold-start
Collaborative filtering	✓	✗
Content-based	✗	✓
Content-aware filtering	✓	✓

Proposed approach

Leverage content information with a clear **music-related** meaning.

Baseline method

Music preference attributes

Experiments

Baseline method

Matrix factorization

Data $\mathbf{Y} = \{y_{u,i}\} \in \mathbb{R}_+^{U \times I}$ = users / items interactions.

- explicit, e.g., likes, ratings...
- implicit, e.g., playcounts.

Matrix factorization

Data $\mathbf{Y} = \{y_{u,i}\} \in \mathbb{R}_+^{U \times I}$ = users / items interactions.

- explicit, e.g., likes, ratings...
- implicit, e.g., playcounts.

Matrix factorization $\mathbf{Y} \approx \mathbf{W}^\mathsf{T}\mathbf{H}$

- $\mathbf{W} = \{w_{k,u}\} \in \mathbb{R}^{K \times U}$: users preferences.
- $\mathbf{H} = \{h_{k,i}\} \in \mathbb{R}^{K \times I}$: songs attributes.

Matrix factorization

Data $\mathbf{Y} = \{y_{u,i}\} \in \mathbb{R}_+^{U \times I}$ = users / items interactions.

- explicit, e.g., likes, ratings...
- implicit, e.g., playcounts.

Matrix factorization $\mathbf{Y} \approx \mathbf{W}^\mathsf{T}\mathbf{H}$

- $\mathbf{W} = \{w_{k,u}\} \in \mathbb{R}^{K \times U}$: users preferences.
- $\mathbf{H} = \{h_{k,i}\} \in \mathbb{R}^{K \times I}$: songs attributes.

Content-aware factorization: $\mathbf{Y} \approx \mathbf{W}^\mathsf{T}\mathbf{H}$ with $\mathbf{h}_i \approx \phi(\mathbf{z}_i)$

- $\mathbf{z}_i$ is a latent content vector extracted from low-level features $\mathbf{x}_i$.

Content-aware weighted matrix factorization [Liang, 2015]

Generative process for the data (binarized playcounts $\mathbf{R}$):

- Observed binarized playcount: $r_{u,i} \sim \mathcal{N}(\mathbf{w}_u^\mathsf{T}\mathbf{h}_i, c_{u,i}^{-1})$.
- User preference factor: $\mathbf{w}_u \sim \mathcal{N}(0, \lambda_W^{-1}\mathbf{I}_K)$.
- Item attribute factor: $\mathbf{h}_i \sim \mathcal{N}(\phi(\mathbf{z}_i), \lambda_H^{-1}\mathbf{I}_K)$.

Liang et al., **Content-Aware Collaborative Music Recommendation Using Pre-trained Neural Networks**, *Proc. of ISMIR*, 2015.

Content-aware weighted matrix factorization [Liang, 2015]

Generative process for the data (binarized playcounts $\mathbf{R}$):

- Observed binarized playcount: $r_{u,i} \sim \mathcal{N}(\mathbf{w}_u^\mathsf{T}\mathbf{h}_i, c_{u,i}^{-1})$.
- User preference factor: $\mathbf{w}_u \sim \mathcal{N}(0, \lambda_W^{-1}\mathbf{I}_K)$.
- Item attribute factor: $\mathbf{h}_i \sim \mathcal{N}(\phi(\mathbf{z}_i), \lambda_H^{-1}\mathbf{I}_K)$.

Content

- Pre-calculate content vector $\mathbf{z}_i \in \mathbb{R}^L$ from low-level features $\mathbf{x}_i$.
- Linear mapping to the item attributes : $\phi(\mathbf{z}_i) = \mathbf{B}\mathbf{z}_i$ with $\mathbf{B} \in \mathbb{R}^{K \times L}$.

Liang et al., **Content-Aware Collaborative Music Recommendation Using Pre-trained Neural Networks**, *Proc. of ISMIR*, 2015.

Content-aware weighted matrix factorization [Liang, 2015]

Generative process for the data (binarized playcounts $\mathbf{R}$):

- Observed binarized playcount: $r_{u,i} \sim \mathcal{N}(\mathbf{w}_u^\mathsf{T}\mathbf{h}_i, c_{u,i}^{-1})$.
- User preference factor: $\mathbf{w}_u \sim \mathcal{N}(0, \lambda_W^{-1}\mathbf{I}_K)$.
- Item attribute factor: $\mathbf{h}_i \sim \mathcal{N}(\phi(\mathbf{z}_i), \lambda_H^{-1}\mathbf{I}_K)$.

Content

- Pre-calculate content vector $\mathbf{z}_i \in \mathbb{R}^L$ from low-level features $\mathbf{x}_i$.
- Linear mapping to the item attributes : $\phi(\mathbf{z}_i) = \mathbf{B}\mathbf{z}_i$ with $\mathbf{B} \in \mathbb{R}^{K \times L}$.

Estimation

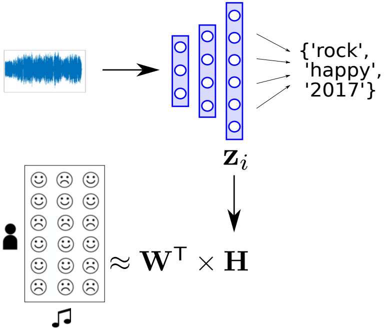
$$\min_{\mathbf{W},\mathbf{H},\mathbf{B}} \sum_{u,i} c_{u,i}(r_{u,i} - \mathbf{w}_u^\mathsf{T}\mathbf{h}_i)^2 + \lambda_W \sum_u ||\mathbf{w}_u||^2 + \lambda_H \sum_i ||\mathbf{h}_i - \mathbf{B}\mathbf{z}_i||^2$$

- Iterative procedure with closed-form updates.

Liang et al., **Content-Aware Collaborative Music Recommendation Using Pre-trained Neural Networks**, *Proc. of ISMIR*, 2015.

Baseline content features

Extract latent factors $\mathbf{z}_i$ from a deep-tagging system.



- ▷ A DNN maps low-level features to tags.
- ▷ $\mathbf{z}_i$ is the last hidden layer output.
  - ✗ No clear meaning of the content feature.

Music preference attributes

The AVD model

Music preference [Fricke, 2019]

Steming from studies in music psychology, musical preference can be conceptualized using three factors:

- *Arousal*: is it energetic/intense or calm?
- *Valence*: is it sad or happy?
- *Depth*: is it "sophisticated" or simple?

Computing AVD

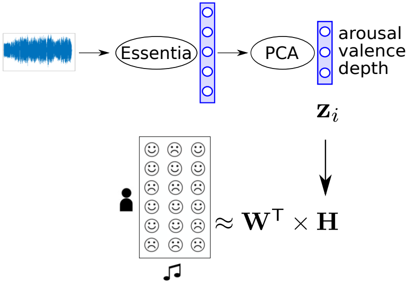
- Collect features using the Essentia toolbox (121 features).
- Keep the most "high-level" ones (17 features).
- PCA on this set of features ($L = 3$) with oblimin rotation.

Fricke et al., **Measuring musical preferences from listening behavior: Data from one million people and 200,000 songs**, *Psychology of Music*, 2019.

AVD - correlations with high-level features

	Arousal	Valence	Depth
Mirex clusters			
1 (Rousing, Passionate)	**0.56**	-0.11	-0.12
2 (Fun, Cheerful)	-0.03	**0.78**	0.01
4 (Humorous, Witty)	-0.05	**0.63**	0.12
5 (Aggressive, Intense)	0.34	**-0.55**	0.30
Mood-related			
Aggressive	**0.63**	-0.48	-0.02
Happy	**0.52**	0.37	-0.36
Party	**0.69**	0.05	0.40
Relaxed	**-0.84**	0.01	0.14
Sad	**-0.80**	0.07	-0.21
Sound-related			
Acoustic	**-0.78**	0.04	-0.25
Average loudness	**0.59**	0.14	-0.07
Danceable	0.23	0.42	**0.52**
Dissonance	**0.86**	-0.03	-0.04
Dynamic complexity	**-0.57**	0.07	0.21
Electronic	0.08	-0.01	**0.74**
Instrumental	-0.35	-0.06	0.23
Tonal	0.04	0.15	**-0.60**

Method



Training

- Extract the AVD factors for all song.
- Incorporate it as content feature $\mathbf{z}_i$ in weighted matrix factorization.
- Train the model to learn $\mathbf{W}$ and $\mathbf{B}$ (and $\mathbf{H}$).

Testing (for cold-start recommendation)

- For a novel song, extract its AVD factors $\mathbf{z}_i$.
- Perform predictions through: $\hat{r}_{u,i} = \mathbf{w}_u^\mathsf{T}\mathbf{B}\mathbf{z}_i$.

Experiments

Protocol

Million song dataset

- ▷ Songs whose Essentia features are available.
- ▷ Filter out inactive user/songs.

# users	$9,132$
# songs	$7,674$
# playcounts	$247,414$
% playcounts	0.35

In-matrix recommendation = traditional collaborative filtering.

- ▷ Keep songs (95 %) for which some listening history is available.
- ▷ In-matrix playcounts split: Train/val/test (70/20/10).

Out-of-matrix recommendation = cold-start scenario.

- ▷ 5 % songs on which the model is not trained (no listening history).

Metric: NDCG (normalized discounted cumulative gain), higher is better.

Results

	In-matrix
Collaborative filtering (no content)	0.35
Proposed (content-aware)	
Essentia (before PCA)	0.36
AVD (after PCA)	0.35

▷ Similar performance for in-matrix recommendation.

Results

	In-matrix	Out-of-matrix
Collaborative filtering (no content)	0.35	–
Pure content (no user similarities)	–	0.19
Proposed (content-aware)		
Essentia (before PCA)	0.36	$\mathbf{0.22}$
AVD (after PCA)	0.35	$\mathbf{0.21}$

- ▷ Similar performance for in-matrix recommendation.
- ▷ The AVD features allows to address the cold-start problem.
- ▷ Content-aware filtering $>$ pure content-based approach.

Conclusion

Music preference attributes are relevant for cold-start recommendation.

Perspectives

- Combination with other types of content (e.g., tags, lyrics).
- Alternative content/attributes mappings (e.g., non-linear, deep), see our extended paper:

P. Magron, C. Févotte, "Neural content-aware collaborative filtering for cold-start music recommendation", 2021, https://arxiv.org/abs/2102.12369

https://github.com/magronp/mus-reco-avd