Similarity of Nearest-Neighbor Query Results in Deep Latent Spaces

Philip Tovstogan **Xavier Serra** **Dmitry Bogdanov**
Music Technology Group, Universitat Pompeu Fabra
`first.last@upf.edu`

ABSTRACT

Music recommendation systems are commonly used for personalized recommendations. However, there are cases where due to privacy concerns or design decisions, there is no user information nor collaborative filtering data available. In those cases, it is possible to use content-based similarity spaces to retrieve the most similar tracks to be recommended based on the reference track. In this paper, we compare the latent spaces extracted from state-of-the-art autotagging models in terms of the similarity between lists of retrieved nearest neighbors. We additionally study item factors from collaborative-filtering data as a reference. We provide insights into how much the choice of the architecture, training dataset, or model layer (output vs. penultimate) as well as a projection of the latent space onto 2D changes the list of retrieved nearest neighbors. We release the dataset of 9 content-based and 3 collaborative-filtering latent representations of 29 275 tracks from Jamendo that we use for the evaluation. Moreover, we perform an online user experiment to compare the perceived track-to-track similarity of the selected evaluated latent spaces. The results show that content-based spaces show better results in our scenario, particularly embeddings from penultimate layers of auto-tagging architectures.

1. INTRODUCTION

In the age of prevalent music streaming, music recommendation systems are currently one of the primary ways for people to listen and find music. While collaborative filtering (CF) approaches are still within the state-of-the-art for personalized music recommendation, pure CF falls short at the cold-start problem and non-personalized recommendations. The content-based (CB) approaches can provide recommendations and suggestions based on item-to-item similarity without CF data, and they are commonly used together with CF in modern recommendation systems [1] to solve the cold-start problem. However, when there is no CF data available due to design decisions or privacy concerns, CB approaches are the only ones that can provide recommendations.

In the domain of music, there are different modalities to the content that can be used for CB approaches. Apart from the audio signal, data that can be used is metadata, user-defined tags, reviews, etc. In this paper, we will focus on audio and current state-of-the-art auto-tagging models. We are interested in how consistent are the latent spaces extracted by auto-tagging models between each other, particularly concerning the choice of the training dataset, architecture, or the layer of the network. These insights can show which variable contributes to the most dissimilar results, which can inform practical decisions on the prioritization of models for A/B testing in an industry scenario with limited resources.

Latent similarity spaces are also quite extensively used in music visualization interfaces [2], where such similarity spaces represent music on a 2D plane or 3D space and facilitate exploration, discovery, and re-discovery of music. The latent spaces usually are high-dimensional, and part of the information is lost by performing the projection. We are interested to see how well the commonly used projection methodologies represent and transform similarity space, and how much of the nearest neighbors' information is preserved.

Furthermore, we investigate how CB approaches compare to CF approaches in a user-less scenario, with CF factors representing a latent similarity space. The motivation is to see if different CB approaches capture more or less of the information that comes from user interactions in CF systems, thus resulting in more or less similar nearest neighbor results.

2. RELATED WORK

Music similarity is a widely researched topic in music information retrieval. In MIREX (Music Information Retrieval Evaluation eXchange) the task of music similarity has been active until 2015, as eventually, the performances of the submitted systems have reached the glass ceiling stemming from evaluation being subjective and limited inter-rater agreement [3]. The music similarity is quite subjective as humans use different dimensions for assessing similarity: genre, moods, tempo, instrumentation, etc. Recent work investigates the importance of inter- and intrarater agreement in the context of music similarity and recommendation [4] that questions the validity of experiments on general music similarity. Thus, is it important to minimize the ambiguity of the evaluation process and provide context or a scenario to allow users to provide more informed answers instead of asking vague questions about which track is more or less similar to the reference track.

In the context of music recommendation, there are many approaches to solve the cold-start problem [1], for example, deep-learning and hybrid approaches [5, 6], or ones trying to predict the CF latent factors from audio [7, 8]. They all attempt to bridge the gap between CF and CB, thus requiring CF data to train the model. In this paper, we consider the scenario without personalization (anonymous user), i.e. where the system has no information about the user and needs to consider only track-to-track similarity.

Among the visualization interfaces of music collections, there are several commonly used techniques to reduce the dimensionality of the original latent spaces [2]. One of the first successful techniques is self-organizing maps (SOM) [9] used in Islands of Music [10] and other works that have followed and were inspired by it. In the more recent works, the newer algorithms such as t-SNE [11] and UMAP [12] gained popularity. They transform space in a non-linear way attempting to capture the relations between individual elements. The classic non-stochastic principal component analysis (PCA) approach can also be used [13]. While it is not as good at capturing the individual relationships between items, it captures the global structure of the whole space.

3. SIMILARITY METRIC

We aim to compare multiple latent spaces that contain the same set of items (music tracks). If we use one track as a reference and retrieve the nearest neighbors to the reference, we would have several different lists of nearest neighbors for each latent space. We introduce a simple metric S_n to calculate the similarity between two ranked lists of nearest neighbors L at the cutoff of n tracks that are obtained from two music similarity spaces X and Y. To differentiate this similarity between spaces from the music similarity that we also talk about, we use the term *NN-similarity* in this paper. We divide the number of tracks that are common in both lists by the cutoff to obtain the value between 0 (no common tracks) and 1 (all tracks are the same):

$$S_n(X, Y) = \frac{|L_{X,n} \cap L_{Y,n}|}{n} \qquad (1)$$

If we consider the following example of $n = 5$ nearest neighbors to the track t_0 in the spaces X and Y, we would calculate the NN-similarity in the following way:

$$\begin{aligned} L_{X,5} &= (t_1, t_2, t_3, t_4, t_5) \\ L_{Y,5} &= (t_2, t_6, t_3, t_7, t_8) \\ L_{X,5} \cap L_{Y,5} &= \{t_2, t_3\} \\ S_5(X, Y) &= 2/5 = 0.4 \end{aligned}$$

S_n does not take into account the ranking: t_2 is ranked higher than t_3 in both $L_{X,5}$ and $L_{Y,5}$, but even if the relative rank would be reversed for $L_{Y,5}$, $S_5(X, Y)$ would still have the same value. In reality, if the cutoff n is much smaller than the number of tracks in the dataset ($n \in \{5, 10, 100, 200\}$), the primary difference between the lists is the number of intersected elements, not what is the difference between their ranks. The only potential benefit of using metrics that take ranks into account is to get the finer difference between lists that have the same amount of common tracks. We tried to use Spearman rank correlation or rank-based overlap (RDO) [14], and these metrics did not provide more information about the difference between pairs compared to simple S_n.[1]

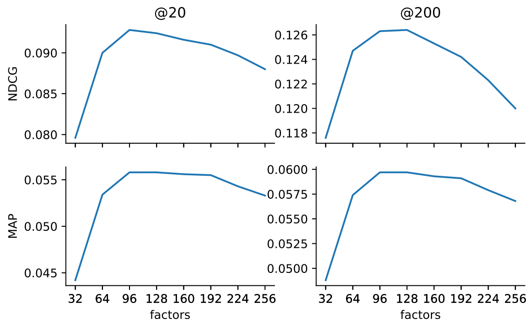


Figure 1. Baseline CF evaluation

4. DATA

Jamendo[2] is a platform that provides royalty-free music for commercial and personal use, including music streaming for venues or video production. In this paper, we use the tracks that are publicly available on their platform under Creative Commons (CC) licenses. In contrast to other datasets that provide audio available under CC licenses (FMA [15]), Jamendo ensures basic technical quality assessment. It was used before in the creation of open music datasets (MTG-Jamendo [16]).

4.1 Collaborative Filtering Features

The collaborative filtering data was provided as part of the collaboration with Jamendo and included 2.2 million interaction events (including plays, skips, etc.) that have associated numeric values assigned via an internal system for 170K tracks and 60K users. We pre-process the data by filtering out the tracks and users that had too few interactions (less than 5) and the top outliers, what results in 31K tracks and 27K users.

We do a pre-analysis of the data to determine the number of factors to be used for the matrix factorization. We use alternating least squares (ALS) algorithm [17] which is one of the SOTA matrix factorization algorithms.[3] Using a stratified split with a test ratio of 0.2 we evaluate different numbers of factors in terms of the performance using normalized discounted cumulative gain (NDCG) and mean average precision (MAP). The results are shown in Figure 1 with 96 factors providing the highest overall performance. We also consider 64 and 128 factors to compare the consistency of several CF spaces.

[1] See additional materials at the companion website philtgun.me/deep-neighbors for reports on other metrics.

[2] jamendo.com

[3] Implementation from github.com/benfred/implicit

Dataset		Architecture	Layer	Dim
MSD	×	MusiCNN	Embeddings	200
			Taggrams	50
MTAT		VGG	Embeddings	256
			Taggrams	50
AudioSet		VGGish	Embeddings	128

Table 1. Dimensions of latent spaces

4.2 Content-based Features

To extract content-based features we use the Essentia library [18] and the following music auto-tagging models [19]:

- *MusiCNN* [20] is a convolutional neural network (CNN) with vertical and horizontal convolutional filter shapes motivated by the music domain. It contains 6 layers and 787 000 trainable parameters.
- *VGG* is an architecture from computer vision [21] based on a deep stack of 3 × 3 convolutional filters that had been adapted for audio [22]. It contains 5 layers and 605 000 trainable parameters.
- *VGGish* is the original implementation of VGG architecture [21] with the number of output units is set to 3087 [23]. The number of trainable parameters is 62 million.

The models provided were pre-trained on several datasets. MusiCNN and VGG have been trained on top 50 tags from Million Song Dataset (MSD) [24] and MagnaTagATune (MTAT) [25]. These datasets contain music and are focused on the music auto-tagging. VGGish has been trained on AudioSet [26] which is an audio event recognition dataset that also includes music. This allows us to compare different architectures that have been trained on the same dataset and the same architecture trained on different datasets.

For the MusiCNN and VGG architectures, we consider the latent spaces constructed by the output layer (taggrams) and penultimate layer (embeddings). VGGish model only provides embeddings. The number of dimensions for the layers is summarized in Table 1. Thus, in total, we extract 9 content-based (CB) feature vectors.

We attempted to process the 31K tracks that we obtained from CF data, but due to some tracks being no longer available or corrupted, this number decreased to 29K.

4.3 Final Dataset

The final large dataset contains 29 275 tracks with successfully extracted CB features. We repeated the matrix factorization on the collaborative filtering data containing only those tracks (29 275 tracks × 27 235 users, 793 963 non-zero values) with the number of factors of 64, 96, and 128 to obtain the CF features. We release this final dataset with 3 CF and 9 CB representations publicly.

We create a smaller subset of the final dataset that is obtained by intersection with MTG-Jamendo [16] test set of split-0 which resulted in 1 372 tracks, which is comparable to a small music collection. We present the experiments on this small dataset, as it visualizes the relative differences between spaces better. [4]

Cutoff	5	10	100	200
Dataset (MSD vs. MTAT)	.26	.26	.46	.56
Arch. (MusiCNN vs. VGG)	.35	.36	.57	.65
Layer (emb. vs. tag.)	.50	.51	.68	.74

Table 2. Average NN-similarity along the variable

5. OFFLINE EXPERIMENTS

5.1 Latent Spaces

We compare the collaborative filtering and content-based spaces that are introduced in Section 4 in terms of NN-similarity S_n that was introduced in Section 3. We present the results using cosine distance to calculate nearest neighbors in Figure 2. Euclidean [5] and cosine distances produce very similar results, except that NN-similarity between CF rankings using Euclidean distance is lower.

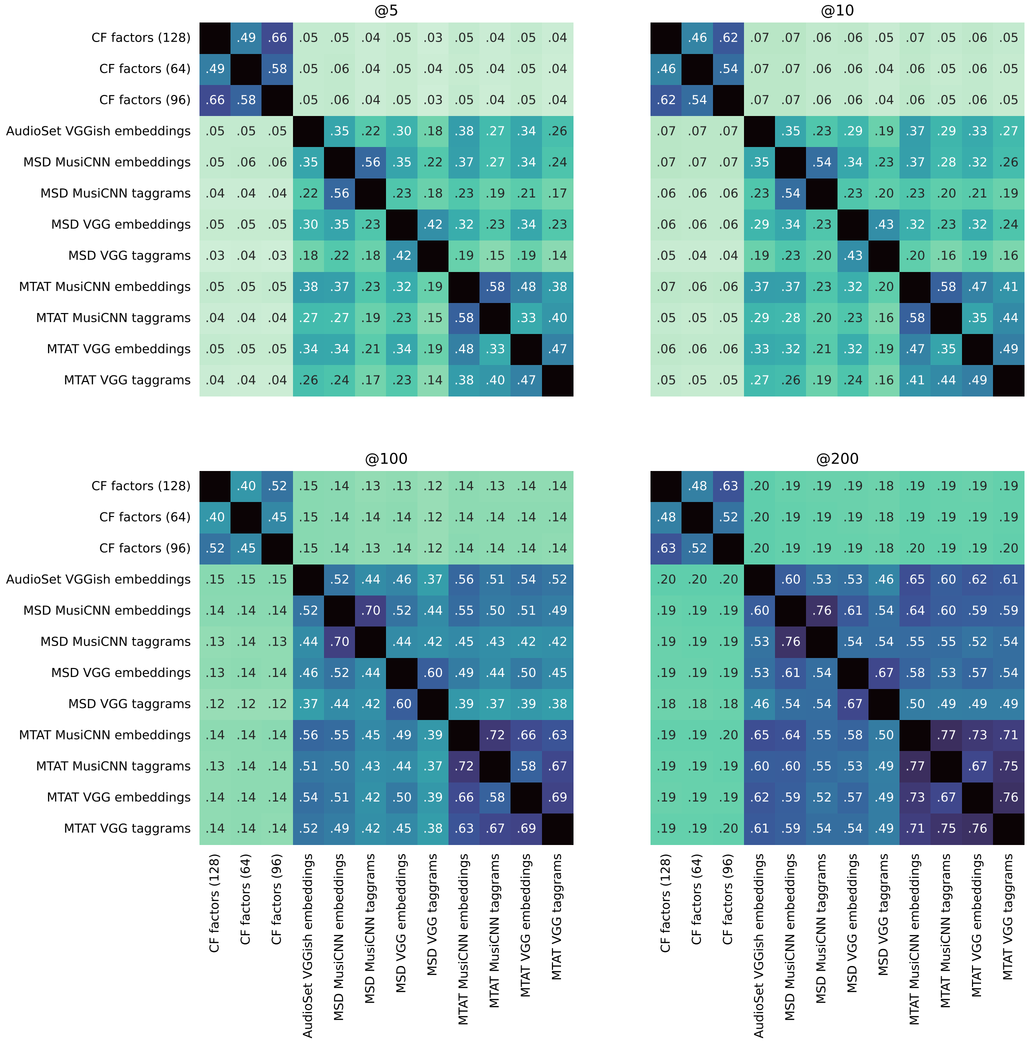
The first thing that stands out in Figure 2 is that the CF spaces are quite dissimilar in terms of NN-similarity from CB spaces, as all pairs of rankings that include CF and CB spaces have the lowest values. This indicates that the music similarity captured by CF and CB spaces is noticeably different. The NN-similarity values between CF spaces stays consistent and is among the highest observed overall at all cutoffs. However, there is enough difference between CF spaces (e.g. max S_5 is 0.66 which means that 2 out of top-5 tracks will be different) to make the number of CF factors an important design decision.

Related to CB embeddings, at smaller cutoffs there is much more variability in the nearest-neighbors lists. Therefore, the choice of the latent space leads to significantly different outcomes in the use-cases that rely on the small number of nearest neighbors. For example, S_{10} varies between 0.16 to 0.58 which means that 4 to 9 tracks will be different between any two CB spaces. At larger cutoffs (100, 200) the NN-similarity between CB spaces is higher (S_{200} ranges from 0.46 to 0.77 between CB spaces).

We can calculate what choice impacts the NN-similarity more: dataset, architecture, or layer. To analyze this, we can fix two out of three variables and calculate the average NN-similarity between the pairs that come from comparing the third variable. For example, to determine how much the choice of *dataset* contributes to NN-similarity, we average the S values of MSD vs. MTAT for MusiCNN embeddings, taggrams, VGG embeddings, and taggrams. As we calculate those for a cutoff value of 5, we get that the average NN-similarity for choice of the dataset is 0.26, which means that if we change the training dataset, roughly only $0.26 \times 5 \approx 1$ track will be the same in the list of 5 nearest neighbors. The values for all cutoff values are presented in Table 2. According to the computed average NN-similarity, latent spaces produced by models trained

[4] The results on the large dataset are available on the companion website.

[5] More figures available on the companion website.



@5

	CF factors (128)	CF factors (64)	CF factors (96)	AudioSet VGGish embeddings	MSD MusiCNN embeddings	MSD MusiCNN taggrams	MSD VGG embeddings	MSD VGG taggrams	MTAT MusiCNN embeddings	MTAT MusiCNN taggrams	MTAT VGG embeddings	MTAT VGG taggrams
CF factors (128)		.49	.66	.05	.05	.04	.05	.03	.05	.04	.05	.04
CF factors (64)	.49		.58	.05	.06	.04	.05	.04	.05	.04	.05	.04
CF factors (96)	.66	.58		.05	.06	.04	.05	.03	.05	.04	.05	.04
AudioSet VGGish embeddings	.05	.05	.05		.35	.22	.30	.18	.38	.27	.34	.26
MSD MusiCNN embeddings	.05	.06	.06	.35		.56	.35	.22	.37	.27	.34	.24
MSD MusiCNN taggrams	.04	.04	.04	.22	.56		.23	.18	.23	.19	.21	.17
MSD VGG embeddings	.05	.05	.05	.30	.35	.23		.42	.32	.23	.34	.23
MSD VGG taggrams	.03	.04	.03	.18	.22	.18	.42		.19	.15	.19	.14
MTAT MusiCNN embeddings	.05	.05	.05	.38	.37	.23	.32	.19		.58	.48	.38
MTAT MusiCNN taggrams	.04	.04	.04	.27	.27	.19	.23	.15	.58		.33	.40
MTAT VGG embeddings	.05	.05	.05	.34	.34	.21	.34	.19	.48	.33		.47
MTAT VGG taggrams	.04	.04	.04	.26	.24	.17	.23	.14	.38	.40	.47	

@10

	CF factors (128)	CF factors (64)	CF factors (96)	AudioSet VGGish embeddings	MSD MusiCNN embeddings	MSD MusiCNN taggrams	MSD VGG embeddings	MSD VGG taggrams	MTAT MusiCNN embeddings	MTAT MusiCNN taggrams	MTAT VGG embeddings	MTAT VGG taggrams
CF factors (128)		.46	.62	.07	.07	.06	.06	.05	.07	.05	.06	.05
CF factors (64)	.46		.54	.07	.07	.06	.06	.04	.06	.05	.06	.05
CF factors (96)	.62	.54		.07	.07	.06	.06	.04	.06	.05	.06	.05
AudioSet VGGish embeddings	.07	.07	.07		.35	.23	.29	.19	.37	.29	.33	.27
MSD MusiCNN embeddings	.07	.07	.07	.35		.54	.34	.23	.37	.28	.32	.26
MSD MusiCNN taggrams	.06	.06	.06	.23	.54		.23	.20	.23	.20	.21	.19
MSD VGG embeddings	.06	.06	.06	.29	.34	.23		.43	.32	.23	.32	.24
MSD VGG taggrams	.05	.04	.04	.19	.23	.20	.43		.20	.16	.19	.16
MTAT MusiCNN embeddings	.07	.06	.06	.37	.37	.23	.32	.20		.58	.47	.41
MTAT MusiCNN taggrams	.05	.05	.05	.29	.28	.20	.23	.16	.58		.35	.44
MTAT VGG embeddings	.06	.06	.06	.33	.32	.21	.32	.19	.47	.35		.49
MTAT VGG taggrams	.05	.05	.05	.27	.26	.19	.24	.16	.41	.44	.49	

@100

	CF factors (128)	CF factors (64)	CF factors (96)	AudioSet VGGish embeddings	MSD MusiCNN embeddings	MSD MusiCNN taggrams	MSD VGG embeddings	MSD VGG taggrams	MTAT MusiCNN embeddings	MTAT MusiCNN taggrams	MTAT VGG embeddings	MTAT VGG taggrams
CF factors (128)		.40	.52	.15	.14	.13	.13	.12	.14	.13	.14	.14
CF factors (64)	.40		.45	.15	.14	.14	.14	.12	.14	.14	.14	.14
CF factors (96)	.52	.45		.15	.14	.13	.14	.12	.14	.14	.14	.14
AudioSet VGGish embeddings	.15	.15	.15		.52	.44	.46	.37	.56	.51	.54	.52
MSD MusiCNN embeddings	.14	.14	.14	.52		.70	.52	.44	.55	.50	.51	.49
MSD MusiCNN taggrams	.13	.14	.13	.44	.70		.44	.42	.45	.43	.42	.42
MSD VGG embeddings	.13	.14	.14	.46	.52	.44		.60	.49	.44	.50	.45
MSD VGG taggrams	.12	.12	.12	.37	.44	.42	.60		.39	.37	.39	.38
MTAT MusiCNN embeddings	.14	.14	.14	.56	.55	.45	.49	.39		.72	.66	.63
MTAT MusiCNN taggrams	.13	.14	.14	.51	.50	.43	.44	.37	.72		.58	.67
MTAT VGG embeddings	.14	.14	.14	.54	.51	.42	.50	.39	.66	.58		.69
MTAT VGG taggrams	.14	.14	.14	.52	.49	.42	.45	.38	.63	.67	.69	

@200

	CF factors (128)	CF factors (64)	CF factors (96)	AudioSet VGGish embeddings	MSD MusiCNN embeddings	MSD MusiCNN taggrams	MSD VGG embeddings	MSD VGG taggrams	MTAT MusiCNN embeddings	MTAT MusiCNN taggrams	MTAT VGG embeddings	MTAT VGG taggrams
CF factors (128)		.48	.63	.20	.19	.19	.19	.18	.19	.19	.19	.19
CF factors (64)	.48		.52	.20	.19	.19	.19	.18	.19	.19	.19	.19
CF factors (96)	.63	.52		.20	.19	.19	.19	.18	.20	.19	.19	.20
AudioSet VGGish embeddings	.20	.20	.20		.60	.53	.53	.46	.65	.60	.62	.61
MSD MusiCNN embeddings	.19	.19	.19	.60		.76	.61	.54	.64	.60	.59	.59
MSD MusiCNN taggrams	.19	.19	.19	.53	.76		.54	.54	.55	.55	.52	.54
MSD VGG embeddings	.19	.19	.19	.53	.61	.54		.67	.58	.53	.57	.54
MSD VGG taggrams	.18	.18	.18	.46	.54	.54	.67		.50	.49	.49	.49
MTAT MusiCNN embeddings	.19	.19	.20	.65	.64	.55	.58	.50		.77	.73	.71
MTAT MusiCNN taggrams	.19	.19	.19	.60	.60	.55	.53	.49	.77		.67	.75
MTAT VGG embeddings	.19	.19	.19	.62	.59	.52	.57	.49	.73	.67		.76
MTAT VGG taggrams	.19	.19	.20	.61	.59	.54	.54	.49	.71	.75	.76	

Figure 2. Nearest neighbor similarity (S_n) of CB vs. CF spaces

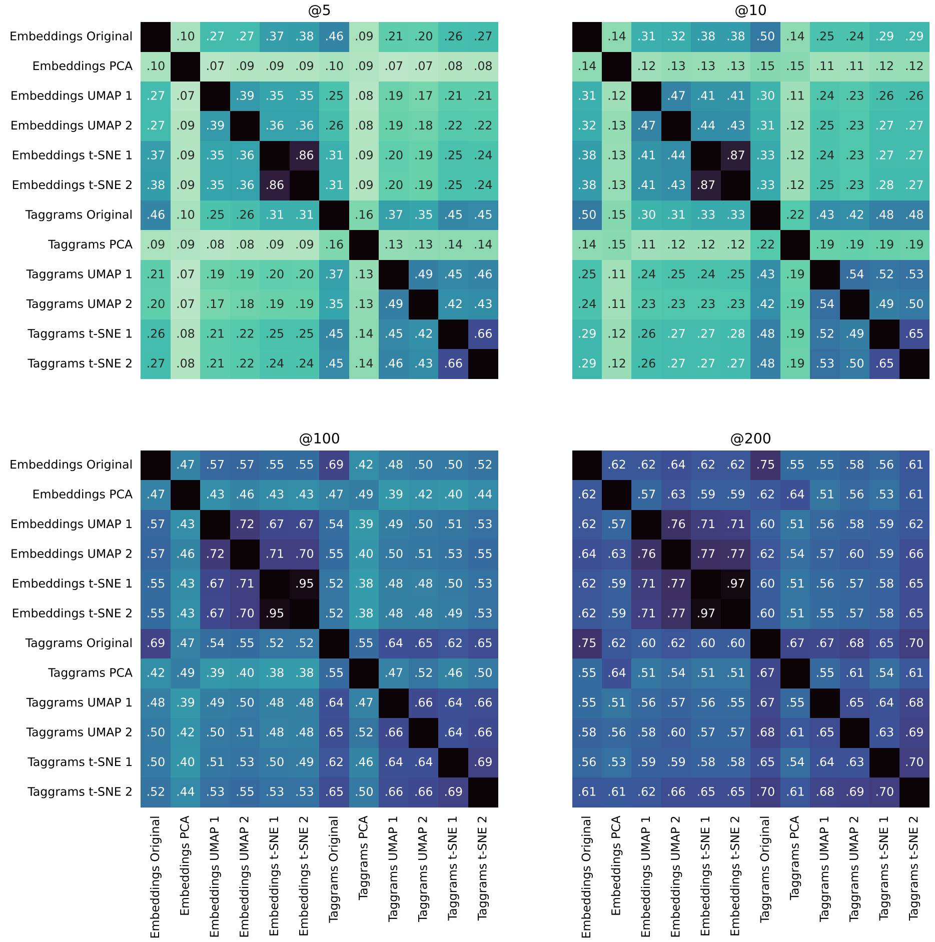
on different datasets (MSD vs. MTAT) are more dissimilar than the ones using different architectures (MusiCNN vs. VGG). Indeed MusiCNN and VGG are both CNN-based and share some similarities.

Regarding the choice of the layer (taggrams vs. embeddings), we can observe the highest NN-similarity when comparing spaces generated by the same model (same dataset and architecture). At the same time, taggram spaces are dissimilar to other CB spaces. That makes sense for spaces from different datasets, as the resulting tag spaces have different vocabulary and semantics. Interestingly enough, it also holds for different architectures on MSD (e.g. MSD MusiCNN vs. VGG taggrams $S_5 = 0.18$ which is close to MSD vs. MTAT MusiCNN taggrams: $S_5 = 0.19$). However, the NN-similarity is much higher for MTAT MusiCNN vs. VGG taggrams: $S_5 = 0.40$.

Overall the MTAT dataset seems to produce spaces that are in general more similar to each other compared to MSD. It can be attributed to the smaller size of MTAT, where the difference between architectures cannot be as pronounced. However, if the tag predictions produced by different architectures on the same dataset are close to each other, that might indicate the quality of annotations. While MSD annotations come from Last.fm folksonomy (every user can assign any tag), MTAT annotations come from the gamified system, where the annotators are encouraged to assign tags that might be similar to ones used by the other people [25].

Another interesting observation is that embeddings of different datasets and architectures, despite having higher dimensionality produce lists of nearest neighbors that are quite similar to each other (minimum values between CB embedding spaces: $S_5 = 0.30$, $S_{10} = 0.29$, $S_{100} = 0.46$, and $S_{200} = 0.53$). This is especially prominent at lower cutoff values (5, 10).

In the context of online evaluation with limited resources, it may be necessary to select a subset of latent spaces. Based on the results from Table 2, it makes sense to prioritize models trained on the different datasets rather than different architectures.



@5	Embeddings Original	Embeddings PCA	Embeddings UMAP 1	Embeddings UMAP 2	Embeddings t-SNE 1	Embeddings t-SNE 2	Taggrams Original	Taggrams PCA	Taggrams UMAP 1	Taggrams UMAP 2	Taggrams t-SNE 1	Taggrams t-SNE 2
Embeddings Original		.10	.27	.27	.37	.38	.46	.09	.21	.20	.26	.27
Embeddings PCA	.10		.07	.09	.09	.09	.10	.09	.07	.07	.08	.08
Embeddings UMAP 1	.27	.07		.39	.35	.35	.25	.08	.19	.17	.21	.21
Embeddings UMAP 2	.27	.09	.39		.36	.36	.26	.08	.19	.18	.22	.22
Embeddings t-SNE 1	.37	.09	.35	.36		.86	.31	.09	.20	.19	.25	.24
Embeddings t-SNE 2	.38	.09	.35	.36	.86		.31	.09	.20	.19	.25	.24
Taggrams Original	.46	.10	.25	.26	.31	.31		.16	.37	.35	.45	.45
Taggrams PCA	.09	.09	.08	.08	.09	.09	.16		.13	.13	.14	.14
Taggrams UMAP 1	.21	.07	.19	.19	.20	.20	.37	.13		.49	.45	.46
Taggrams UMAP 2	.20	.07	.17	.18	.19	.19	.35	.13	.49		.42	.43
Taggrams t-SNE 1	.26	.08	.21	.22	.25	.25	.45	.14	.45	.42		.66
Taggrams t-SNE 2	.27	.08	.21	.22	.24	.24	.45	.14	.46	.43	.66	

@10	Embeddings Original	Embeddings PCA	Embeddings UMAP 1	Embeddings UMAP 2	Embeddings t-SNE 1	Embeddings t-SNE 2	Taggrams Original	Taggrams PCA	Taggrams UMAP 1	Taggrams UMAP 2	Taggrams t-SNE 1	Taggrams t-SNE 2
Embeddings Original		.14	.31	.32	.38	.38	.50	.14	.25	.24	.29	.29
Embeddings PCA	.14		.12	.13	.13	.13	.15	.15	.11	.11	.12	.12
Embeddings UMAP 1	.31	.12		.47	.41	.41	.30	.11	.24	.23	.26	.26
Embeddings UMAP 2	.32	.13	.47		.44	.43	.31	.12	.25	.23	.27	.27
Embeddings t-SNE 1	.38	.13	.41	.44		.87	.33	.12	.24	.23	.27	.27
Embeddings t-SNE 2	.38	.13	.41	.43	.87		.33	.12	.25	.23	.28	.27
Taggrams Original	.50	.15	.30	.31	.33	.33		.22	.43	.42	.48	.48
Taggrams PCA	.14	.15	.11	.12	.12	.12	.22		.19	.19	.19	.19
Taggrams UMAP 1	.25	.11	.24	.25	.24	.25	.43	.19		.54	.52	.53
Taggrams UMAP 2	.24	.11	.23	.23	.23	.23	.42	.19	.54		.49	.50
Taggrams t-SNE 1	.29	.12	.26	.27	.27	.28	.48	.19	.52	.49		.65
Taggrams t-SNE 2	.29	.12	.26	.27	.27	.27	.48	.19	.53	.50	.65	

@100	Embeddings Original	Embeddings PCA	Embeddings UMAP 1	Embeddings UMAP 2	Embeddings t-SNE 1	Embeddings t-SNE 2	Taggrams Original	Taggrams PCA	Taggrams UMAP 1	Taggrams UMAP 2	Taggrams t-SNE 1	Taggrams t-SNE 2
Embeddings Original		.47	.57	.57	.55	.55	.69	.42	.48	.50	.50	.52
Embeddings PCA	.47		.43	.46	.43	.43	.47	.49	.39	.42	.40	.44
Embeddings UMAP 1	.57	.43		.72	.67	.67	.54	.39	.49	.50	.51	.53
Embeddings UMAP 2	.57	.46	.72		.71	.70	.55	.40	.50	.51	.53	.55
Embeddings t-SNE 1	.55	.43	.67	.71		.95	.52	.38	.48	.48	.50	.53
Embeddings t-SNE 2	.55	.43	.67	.70	.95		.52	.38	.48	.48	.49	.53
Taggrams Original	.69	.47	.54	.55	.52	.52		.55	.64	.65	.62	.65
Taggrams PCA	.42	.49	.39	.40	.38	.38	.55		.47	.52	.46	.50
Taggrams UMAP 1	.48	.39	.49	.50	.48	.48	.64	.47		.66	.64	.66
Taggrams UMAP 2	.50	.42	.50	.51	.48	.48	.65	.52	.66		.64	.66
Taggrams t-SNE 1	.50	.40	.51	.53	.50	.49	.62	.46	.64	.64		.69
Taggrams t-SNE 2	.52	.44	.53	.55	.53	.53	.65	.50	.66	.66	.69	

@200	Embeddings Original	Embeddings PCA	Embeddings UMAP 1	Embeddings UMAP 2	Embeddings t-SNE 1	Embeddings t-SNE 2	Taggrams Original	Taggrams PCA	Taggrams UMAP 1	Taggrams UMAP 2	Taggrams t-SNE 1	Taggrams t-SNE 2
Embeddings Original		.62	.62	.64	.62	.62	.75	.55	.55	.58	.56	.61
Embeddings PCA	.62		.57	.63	.59	.59	.62	.64	.51	.56	.53	.61
Embeddings UMAP 1	.62	.57		.76	.71	.71	.60	.51	.56	.58	.59	.62
Embeddings UMAP 2	.64	.63	.76		.77	.77	.62	.54	.57	.60	.59	.66
Embeddings t-SNE 1	.62	.59	.71	.77		.97	.60	.51	.56	.57	.58	.65
Embeddings t-SNE 2	.62	.59	.71	.77	.97		.60	.51	.55	.57	.58	.65
Taggrams Original	.75	.62	.60	.62	.60	.60		.67	.67	.68	.65	.70
Taggrams PCA	.55	.64	.51	.54	.51	.51	.67		.55	.61	.54	.61
Taggrams UMAP 1	.55	.51	.56	.57	.56	.55	.67	.55		.65	.64	.68
Taggrams UMAP 2	.58	.56	.58	.60	.57	.57	.68	.61	.65		.63	.69
Taggrams t-SNE 1	.56	.53	.59	.59	.58	.58	.65	.54	.64	.63		.70
Taggrams t-SNE 2	.61	.61	.62	.66	.65	.65	.70	.61	.68	.69	.70	

Figure 3. Nearest neighbor similarity (S_n) of different projections of MSD MusiCNN embeddings and taggrams

5.2 Projections

One of the applications of the latent spaces is to visualize the similarity between tracks. Hence, we want to use the same methodology to compare how well the NN-similarity is preserved while being projected on a 2D plane. Although some exploration interfaces use 3D planes, for consistency with previous research on music exploration [27] we only consider 2D. We consider PCA [28], t-SNE [11] and UMAP [12]. Moreover, as t-SNE and UMAP are stochastic, we consider two different seeds for each to measure the robustness. From previously obtained results it doesn't make sense to compare projections for different datasets or architectures, however, as embeddings and taggrams are quite similar to each other we consider both of them. Figure 3 shows the results for MSD MusiCNN embeddings and taggrams using Euclidean distance to calculate nearest neighbors, as it firstly, makes more sense to use in 2D, and secondly, cosine distance similarity is significantly lower for most pairs. [6]

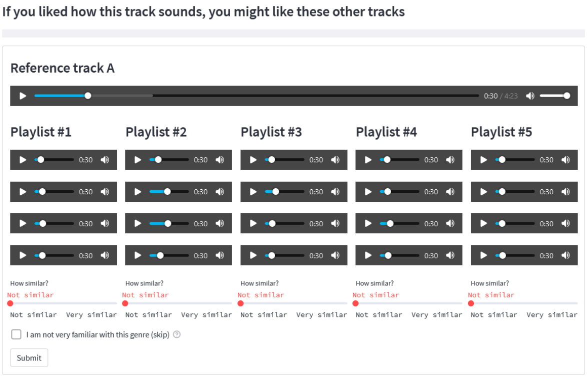
Comparing different projections, it is evident from Figure 3 that t-SNE exhibits the highest NN-similarity to the original spaces. Nevertheless, this projection leads to noticeable changes in rankings (e.g. $S_{10} = 0.42$ for embeddings means that 6 tracks in the top-10 list will be different on average). UMAP is the second-best projection, with PCA being the poorest at preserving NN-similarity. The NN-similarity between different seeds for t-SNE is quite close to 1.0, which means that it is also more robust than UMAP. In general, at larger cutoff values the NN-similarity values are closer to each other. As PCA is a linear projection that works well in preserving the global structure of data without much consideration for nearest neighbors, its NN-similarity is quite low for small cutoff values (5, 10).

From a practical perspective, we see that using t-SNE for projection provides the best results and preserves more than 40% of nearest neighbors for small cutoffs. That means that in the visualization interface, among the 5 closest tracks in projected space 2 tracks are also closest in the original space to the reference track.

6. ONLINE EXPERIMENTS

Even if music similarity is quite subjective, it can be partially alleviated by asking more specific questions to the participants, as shown in related work in Section 2. We use a methodology similar to [29] to evaluate which spaces provide a better representation of music similarity for mu-

[6] Figures of cosine distance is available at companion website.



If you liked how this track sounds, you might like these other tracks

Reference track A

Playlist #1 | Playlist #2 | Playlist #3 | Playlist #4 | Playlist #5

How similar? Not similar — Not similar … Very similar (×5)

□ I am not very familiar with this genre (skip)

Submit

Figure 4. Online experiment interface

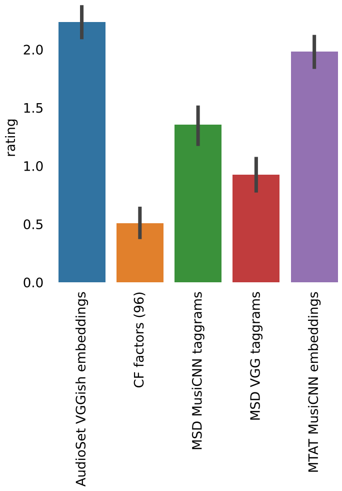


Figure 5. Online experiment results

sic recommendation. The difference with other studies is that this methodology evaluates the perception of playlists of top-N similar tracks, instead of individual comparisons of pairs of tracks. This approach is better aligned with tasks of music exploration and playlist generation.

We use the same small dataset for this experiment for consistency with offline experiments. The participants are presented with a reference track and several candidate playlists containing the nearest neighbors (ordered by their similarity to the reference) for each latent space. They are asked to rate the similarity of each playlist to the reference track in the hypothetical scenario of music recommendation: “If you liked how this track sounds, you might like these other tracks”. The order of reference tracks is the same for all participants, while the playlists are presented in random order.

As the number of choices that can be presented to participants is limited by possible cognitive overload, we selected the 5 most dissimilar latent spaces: CF 96, MSD VGG taggrams, MSD MusiCNN taggrams, MTAT MusiCNN embeddings, and VGGish embeddings. We randomly select 4 reference tracks ensuring that they are quite different from each other and span several genres. In the interest of keeping the time to complete one instance of the experiment as low as possible while providing enough information to the participants, we decided to include 4 tracks in each playlist making it 21 tracks per reference track (including the latter) and 84 tracks in total. Because asking participants to listen to each track completely is unreasonable, by default we present the participant with a segment of 15 seconds that starts at 0:30 and ends at 0:45. However, the participants can use the controls of each player to play more different sections of the track if they feel that they need more information. Participants are encouraged to not spend much time on each track and use their intuition to rate the similarity, what is communicated explicitly in the instructions to the experiment. To measure the perceived similarity we provide a slider that uses a 4-point Likert scale: 0 - not similar, 1 - somewhat similar, 2 - quite similar, and 3 - very similar. We specifically avoided the neutral option to force participants to give their opinion. The interface of the experiment is shown in Figure 4.

We provide introductory text that explains the experiment, interface, purpose and allows the participants to continue with the experiment once they give their explicit consent. After circulating the link to the experiment [7] in the relevant communities (mailing lists, Twitter, subreddits [8]) we obtained data from 39 participants. We asked optional general demographic questions to verify coverage of different demographic groups. All participants are aged from 14–64 with the majority (53%) falling into the age group of 25–34. 55% of participants identify themselves as men, 33% as women, 9% non-binary and 3% preferred not to say. Concerning music background, 18% don’t have any music training, 42% have some, 37% are hobbyists, and 3% (1) are professional musicians. The majority of participants (52%) listen to music on average 2–3 hours per day with the whole population listening from less than 1 hour up to 6–7 hours per day.

We use the Shapiro test (p-value<0.001) to verify the assumptions for the ANOVA test. We perform ANOVA (p-value<0.001) and Kruskal-Wallis (p-value<0.001) tests to verify if the choice of the latent space makes the similarity results significantly different. Subsequently, we use Tukey’s honestly significantly differenced (HSD) test to identify which pairs of latent spaces are significantly different. The only pair of spaces where the difference is not significant is AudioSet VGGish vs. MTAT MusiCNN embeddings (p-value of 0.07).

Figure 5 shows the average ranking performance of each latent space with the standard deviation represented as a vertical line. We can see as both embedding spaces (AudioSet VGGish, MTAT MusiCNN) that we have chosen for the online experiment perform the best, with no statistical difference between them. An interesting observation is that AudioSet is a generic audio event recognition dataset, and the VGGish model trained on it performs comparably to the embeddings from the music auto-tagging dataset MTAT. MSD MusiCNN taggram space has a worse average similarity rating, with MSD VGG taggrams following

[7] `philtgun.me/similarity-experiment`

[8] `reddit.com/r/samplesize`

it. The poor performance of CF factors space can be attributed to the mismatch of the use-case, as it is intended to be used in conjunction with the user factors, not as a latent space. It is also possible that the small size of the dataset impacted the poor performance of CF factors.

The results show that content-based latent spaces can power the anonymous recommendation systems with a similarity that is rated at least as *quite similar*. This is a positive takeaway for exploration and visualization systems that can be built on top of similarity latent spaces.

7. CONCLUSIONS

We compared different collaborative filtering and content-based latent spaces in terms of the nearest-neighbor similarity. We observed that nearest neighbors obtained from CF spaces are very dissimilar to nearest neighbors obtained from CB approaches. Focusing on CB spaces, we identified that the choice of the training dataset (MSD vs. MTAT) tends to produce the most dissimilar spaces, followed by the architecture, and then layer. We observed that taggram spaces tend to be dissimilar across different datasets and architecture, while embedding spaces tend to be more similar. Interestingly, the consistency of CB latent spaces derived from a dataset may differ in terms of their nearest-neighbors similarity, as we observed on the example of the MTAT vs. MSD datasets. In the context of 2D visualization of latent spaces, t-SNE exhibits the highest nearest-neighbors similarity between original and projected spaces.

We performed an online experiment to evaluate a selection of dissimilar latent spaces in the context of music similarity for music recommendation. The results show that the CB spaces can be successfully used in music recommendation/exploration scenarios where user-generated data is absent due to design decisions. We observe that embedding spaces (AudioSet VGGish, MTAT MusiCNN) perform significantly better than taggram spaces (MSD MusiCNN, MSD VGG).

All analysis [9] and experiment interface [10] code is publicly available on GitHub, under Apache 2.0 license. The latent spaces are published on Zenodo [11] under CC BY-NC-SA 4.0 license, and the audio for the small dataset is available in MTG-Jamendo dataset. [12]

Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 765068.

This work was conducted partially during the internship in Jamendo as part of the MIP-Frontiers project. We thank Jamendo for the collaboration, providing collaborative filtering data, and facilitation of the audio processing.

[9] `github.com/philtgun/compare-embeddings`
[10] `github.com/philtgun/similarity-experiment`
[11] `zenodo.org/record/6010468`
[12] `mtg.github.io/mtg-jamendo-dataset`

8. REFERENCES

[1] F. Ricci, L. Rokach, and B. Shapira, Eds., *Recommender Systems Handbook*, 3rd ed. Springer, Mar. 2022, in press.

[2] P. Knees, M. Schedl, and M. Goto, "Intelligent user interfaces for music discovery," *Transactions of the International Society for Music Information Retrieval*, vol. 3, no. 1, pp. 165–179, Oct. 2020.

[3] A. Flexer, "On inter-rater agreement in audio music similarity," in *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR)*. Taipei, Taiwan: Zenodo, Oct. 2014, pp. 245–250.

[4] A. Flexer, T. Lallai, and K. Rašl, "On evaluation of inter- and intra-rater agreement in music recommendation," *Transactions of the International Society for Music Information Retrieval*, vol. 4, no. 1, p. 182, Nov. 2021.

[5] S. Oramas, O. Nieto, M. Sordo, and X. Serra, "A deep multimodal approach for cold-start music recommendation," in *Proceedings of the 2nd Workshop on Deep Learning for Recommender Systems (DLRS)*. Como, Italy: ACM, Aug. 2017, pp. 32–37.

[6] X. Wang and Y. Wang, "Improving content-based and hybrid music recommendation using deep learning," in *Proceedings of the 22nd ACM International Conference on Multimedia (MM)*. Orlando, FL, USA: ACM, Nov. 2014, pp. 627–636.

[7] A. van den Oord, S. Dieleman, and B. Schrauwen, "Deep content-based music recommendation," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 26. Lake Tahoe, NV, United States: Curran Associates, Inc., Dec. 2013, pp. 2643–2651.

[8] A. Ferraro, Y. Kim, S. Lee, B. Kim, N. Jo, S. Lim, S. Lim, J. Jang, S. Kim, X. Serra, and D. Bogdanov, "Melon playlist dataset: A public dataset for audio-based playlist generation and music tagging," in *2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Toronto, ON, Canada: IEEE, Jun. 2021, pp. 536–540.

[9] T. Kohonen, *Self-organizing maps*, 3rd ed., ser. Springer Series in Information Sciences. Berlin, Heidelberg: Springer, 2001, vol. 30.

[10] E. Pampalk, A. Rauber, and D. Merkl, "Content-based organization and visualization of music archives," in *Proceedings of the 10th ACM International Conference on Multimedia (MM)*. Juan-les-Pins, France: ACM, Dec. 2002, pp. 570–579.

[11] L. v. d. Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2605, Nov. 2008.

[12] L. McInnes, J. Healy, and J. Melville, "UMAP: uniform manifold approximation and projection for dimension reduction," Feb. 2018.

[13] D. Smilkov, N. Thorat, C. Nicholson, E. Reif, F. B. Viégas, and M. Wattenberg, "Embedding projector: Interactive visualization and interpretation of embeddings," Dec. 2016.

[14] W. Webber, A. Moffat, and J. Zobel, "A similarity measure for indefinite rankings," *ACM Transactions on Information Systems*, vol. 28, no. 4, pp. 1–38, Nov. 2010.

[15] M. Defferrard, K. Benzi, P. Vandergheynst, and X. Bresson, "FMA: A dataset for music analysis," in *Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR)*. Suzhou, China: Zenodo, Oct. 2017, pp. 316–323.

[16] D. Bogdanov, M. Won, P. Tovstogan, A. Porter, and X. Serra, "The MTG-Jamendo dataset for automatic music tagging," in *Machine Learning for Music Discovery Workshop (ML4MD), International Conference on Machine Learning (ICML)*, Long Beach, CA, USA, Jun. 2019.

[17] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *2008 Eighth IEEE International Conference on Data Mining (ICDM)*. Pisa, Italy: IEEE, Dec. 2008, pp. 263–272.

[18] D. Bogdanov, N. Wack, E. Gómez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J. R. Zapata, and X. Serra, "Essentia: An audio analysis library for music information retrieval," in *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR)*. Curitiba, Brazil: Zenodo, Nov. 2013, pp. 493–498.

[19] P. Alonso-Jiménez, D. Bogdanov, J. Pons, and X. Serra, "Tensorflow audio models in Essentia," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Barcelona, Spain: IEEE, May 2020, pp. 266–270.

[20] J. Pons and X. Serra, "MusiCNN: Pre-trained convolutional neural networks for music audio tagging," Sep. 2019.

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations (ICLR)*. San Diego, CA, USA: arXiv, May 2015.

[22] K. Choi, G. Fazekas, and M. B. Sandler, "Automatic tagging using deep convolutional neural networks," in *Proceedings of the 17th International Society for Music Information Retrieval Conference (ISMIR)*. New York City, NY, USA: Zenodo, Aug. 2016, pp. 805–811.

[23] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. Wilson, "CNN architectures for large-scale audio classification," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. New Orleans, LA, USA: IEEE, Mar. 2017, pp. 131–135.

[24] T. Bertin-Mahieux, D. P. W. Ellis, B. Whitman, and P. Lamere, "The million song dataset," in *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR)*. Miami, FL, USA: Zenodo, Oct. 2011, pp. 591–596.

[25] E. Law, K. West, M. I. Mandel, M. Bay, and J. S. Downie, "Evaluation of algorithms using games: the case of music tagging," in *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*. Kobe, Japan: Zenodo, Oct. 2009, pp. 387–392.

[26] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio Set: An ontology and human-labeled dataset for audio events," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. New Orleans, LA, USA: IEEE, Mar. 2017, pp. 776–780.

[27] P. Tovstogan, X. Serra, and D. Bogdanov, "Web interface for exploration of latent and tag spaces in music auto-tagging," in *Machine Learning for Music Discovery Workshop (ML4MD), International Conference on Machine Learning (ICML)*, Vienna, Austria, Jul. 2020.

[28] K. Pearson, "LIII. On lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, Nov. 1901.

[29] D. Bogdanov, J. Serrà, N. Wack, and P. Herrera, "From low-level to high-level: Comparative study of music similarity measures," in *2009 11th IEEE International Symposium on Multimedia (ISM)*. San Diego, CA, USA: IEEE, Dec. 2009, pp. 453–458.