Content-based Music Recommendation System

Aldiyar Niyazov
St.Petersburg State University
St.Petersburg, Russia
aldiyar.niyazov0320@gmail.com

Elena Mikhailova
ITMO University,
St.Petersburg State University
St.Petersburg, Russia
e.mikhaylova@spbu.ru

Olga Egorova
ITMO University
St.Petersburg, Russia
olva.egorova@gmail.com

***Abstract*—Building a music recommendation system is one of information retrieval tasks. This research is devoted to a content-based music recommender system. The main peculiarity of our work is that the developed recommender system is based on the acoustic similarity of musical compositions. Two approaches of building a content-based music recommender system are considered in this paper. The first is a quite common approach that uses acoustic features analysis. The second approach includes deep learning and computer vision methods application aimed at improving the results of the recommender system.**

I. INTRODUCTION

Nowadays the amount of information provided to the user by modern information systems considerably exceeds the amount, that humans can physically examine, evaluate, and find something that suites to their goals. Therefore, when we face with information services containing a huge amount of content, it is worth to build a recommendation system to solve this problem. The task of the recommender system is to filter all available content and to offer to users only the appropriate information that they are really interested in. There are two main approaches to content filtering: collaborative and content-based.

The main idea of collaborative filtering is the assumption that "similar" users act the same way in similar situations. In accordance with this, a recommender system based on collaborative filtering considers either the actions of other users that are in some kind similar to the user of interest, or the history of the user's ratings of similar content.

Recommender systems based on content filtering method offer such objects to a user, that are similar to the objects having interested him previously. In this case, in contrast to collaborative filtering, the similarity between objects is evaluated not based on user actions, but on the characteristics of the objects themselves. Content filtering relies on the objective characteristics of the objects only and does not depend on the subjective ratings of users.

Building a music recommender system is considered as one of the information retrieval tasks [1]. The user has got a certain information demand that need to be satisfied. Within the framework of a music recommender system this task becomes more complicated, as the user often does not clearly understand what music he wants to listen to. In the modern music services, the collaborative filtering approach is the most common and provides an acceptable quality of recommendations. However, this approach does not work well when a new user or a new music appears, as there is no sufficient information about the user's interaction with the content in such case. This problem is known as the "cold start" problem. The content-based recommender system deals with this task successfully.

The object of our research is a content-based music recommender system. The main idea of our work is that the developed recommender system is based not on the external features of musical compositions, such as genre, artist, title, tags, etc., but on the acoustic similarity of musical compositions.

To solve the "cold start" problem we need to find the similar sounding music. For example, to introduce listeners to the music of an unpopular artist, we can recommend his music along with similar sounding compositions of more popular artists. Therefore, to evaluate the quality of our recommendations, we measure how accurately our recommender system determines the music similarity.

II. RELATED WORKS

Peter Knees and Markus Schedl described the main tasks of music information analysis in their book [1]. In particular, the authors conducted a very detailed study of methods determining the musical compositions similarity. They thoroughly described physical properties of acoustic characteristics as well as influence of the latter on human music perception.

Aaron van den Oord and co-authors [2] suggested using spectrograms to deal with the problems of music recommender systems. This allowed them to shift from the acoustic characteristics analysis to computer vision methods application.

Rui Lu and co-authors [3] proposed an artificial neural network architecture, namely, Triplet MatchNet, which was trained to directly detect the acoustic similarity of music. This architecture is based on residual blocks with shortcut connections. Expert assessment was used as a quantitative similarity measure of music sounding.

The authors of the article published in 2020 [4] trained a fully connected neural network to search for the songs written by the same artist. The network was also trained on triplets, but instead of the song spectrograms, the acoustic characteristics of music were used as the input information.

III. METHOD DESCRIPTION

Any acoustic signal including music is digitally represented as a sequence of values taken at a certain sampling frequency. For music compositions, the most popular sampling rates are

22050 Hz and 44100 Hz. That is, one second of an audio recording is encoded by several dozens of thousands of samples. Because of the limited computational capabilities, it is very expensive to analyze signals in such a form. Therefore, we needed to transform the original large dimensional sequence into a more convenient form, but with minimal information loss.

In a rough generalization, we can split our task into two main subtasks:

- To form an adequate representation of a musical composition in a vector space of a certain dimensionality;
- To evaluate the similarity of vector representations of songs.

In the current study, we followed the approach when vector representation of a musical composition is formed by means of the extraction of some acoustic characteristics from the audio signal. Besides, we also tried to improve the obtained results by using artificial neural networks. The similarity of music compositions was determined using distance metrics between the song vector representations. The list of recommendations consisted of 10 nearest neighbors of the test song selected in the vector space of the training sample. The process of recommendations building used in our study is shown in Fig. 1.

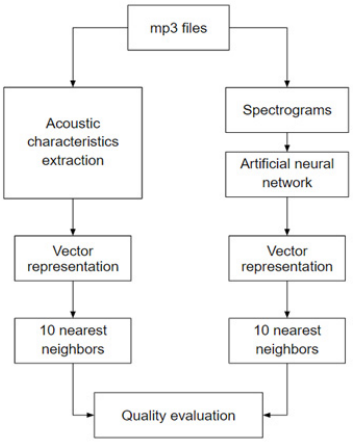

Fig. 1. Recommendation building scheme

IV. Common approach

The main idea of the method [1] is that the original signal is split into overlapping frames that include a certain number of samples (the number of samples depends on the sampling frequency of the signal). By implementing the Fourier transform to each frame, we obtain the spectrum of the signal. Based on the spectrum shape, we compute descriptive statistics and other characteristics, and aggregate them over the entire number of frames (we compute the average, minimum, maximum, standard deviation etc.). As a result, each music composition is mapped into a vector space of acoustic characteristics, the dimensionality of which is equal to 77 features.

In general, the process of recommendations building based on the acoustic characteristics extraction can be represented as follows:

- A certain music composition is given to the input of the recommender system;
- Acoustic characteristics are extracted;
- A feature vector for the composition is built;
- Vector similarity is estimated;
- A list of recommendations is formed.

To extract acoustic features from audio recordings, we use an open-source library Essentia [5]. The characteristics [1] presented in Table I were extracted.

TABLE I. Acoustic features extracted

Feature	Description
spectral centroid	The signal spectrum center. It is computed as the weighted average of the frequencies that are present in the signal. Characterizes the timbre.
spectral kurtosis	The kurtosis coefficient. Characterizes the shape of the signal spectrum.
spectral skewness	The skewness coefficient. Characterizes the shape of the signal spectrum.
spectral spread	The spread. Characterizes the shape of the signal spectrum.
spectral rolloff	Is defined as the relative frequency value within the limits of which a certain part of the spectrum total energy is concentrated. Characterizes the differences between a noise-like signal and a harmonic one.
spectral flux	The spectral flux. Reflects how fast the energy of the spectrum changes, is computed basing on the spectra of the current and the previous frames: the second norm (Euclidean distance) between two normalized spectra.
spectral complexity	The spectral complexity is based on the number of peaks in the input spectrum.
spectral entropy	By splitting each frame into a set of subframes, the energy set for each subframe is calculated. Further, normalizing the energy of each of the subframes by the energy of the entire frame, we can consider the set of energies as a set of probabilities, and calculate the information entropy. Characterizes the differences between voice-like and non-voice-like signals.
zero crossing rate	The number of intersections of the time axis in the audio signal. Characterizes noise-like signals
melbands crest	It is defined as the ratio of the maximum value to the average value of the signal mel-frequency array.
melbands flatness	Reflects the deviation of the signal spectrum power from the flat shape. From the point of view of human perception, it characterizes how tone-like the audio signal is.
pitch salience	The pitch salience is defined as the ratio of the maximum value of the spectrum autocorrelation to the unbiased autocorrelation value. It is a quick measure of tone perception. Sounds without tones (sound effects without tones) and pure tones have an average pitch value close to 0, whereas sounds having multiple harmonics in the spectrum have a higher pitch value.
chord stability	The stability of the chords
hpcp crest	Harmonic tone class
average loudness	Average sound loudness
dynamic complexity	It is defined as the average absolute deviation from the global volume level estimate according to the dB scale. It is related to the dynamic range and the amount of volume fluctuations present in the recording.
beats count	The number of (rhythmic) beats
bpm	Beats per minute
onset rate	Sound onset rate
danceability	The music rhythmicity
chords changes rate	The rate of chords changes
chords number rate	The number of chords

To determine the similarity between vector representations, the following distance metrics were used: Euclidean, Manhattan, and cosine distances [6]. The most relevant recommendations were obtained using the cosine distance to calculate the similarity of the vectors. To make sure that our model is relevant, we compared it with a recommender system that gives random recommendations, and with a recommender system that gives random recommendations under the condition that they are of the same genre with the target track. The results are shown in Table II.

V. NEURAL NETWORK APPLICATION

Nowadays, artificial neural networks (ANN) are quite successful in solving different tasks in the field of image processing (computer vision) and automatic text processing (natural language processing). Therefore, we decided to use acoustic signal spectrograms as graphical representations of the audio signal, and then to determine the similarity of musical compositions using computer vision methods [7].

A spectrogram is a matrix of values that describe the signal spectral power density depending on the time. The most common representation of a spectrogram is a two-dimensional chart, where the horizontal axis represents time, and the vertical axis represents frequency; the third dimension is represented by the intensity or the color of each image point. That is, the spectrogram is a matrix, in which the cells represent the intensity values of a certain frequency at a certain point in time.

Human sound perception is nonuniform. Our ear is designed in such a way that we are more sensitive to changes in frequency in the low frequency region than to those in the high frequency region. Therefore, when solving tasks related to the human sound perception, it is common practice to change the frequency measurement scale from Hz to Mel (1). For this reason, we used mel-frequency spectrograms to display the signal.

$$mel = 2595\, log_{10}\left(1 + \frac{f}{700}\right) \quad (1)$$

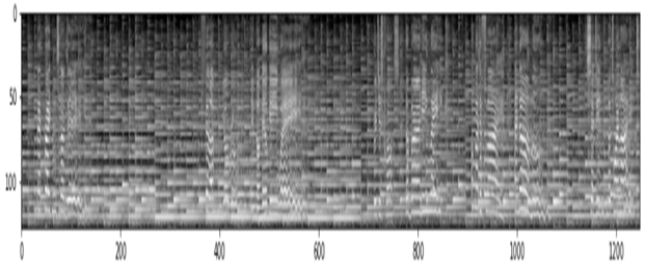

Fig. 2. Mel frequency spectrogram of a musical composition

Traditionally, computer vision problems were initially solved by using convolutional neural networks (CNN). There are various CNN architectures, but usually they all consist of two main parts:

1) Feature extractor block;
2) Classification block.

The feature extractor block consisting of convolutional layers transforms the input image into a vector representation that stores information about the image. Then, the classification block maps the resulting vector to one of the classes.

In 2014, a group of researchers from Google proposed a method for image ranking based on their similarity [8]. In relation to our task, we can directly train the ANN to find the similarity measure between the audio signal spectrograms. To train the ANN, three spectrograms (a triplet) are given to the input of the ANN: the target song, a song similar to the target one and a song that is different from the target one. Analyzing these triplets, the ANN is trained to match a vector from some vector space to a spectrogram in such a way that the distance between vectors of similar-sounding music is less than the distance between the vectors of differing music (2).

$$D\left(f(p_i), f(p_i^+)\right) < D\left(f(p_i), f(p_i^-)\right) \quad (2)$$

where D is the distance measure between the vectors, f is the function of spectrogram mapping to the vector space and p is a spectrogram.

Triplet Margin Loss (3) is used as a loss function. During the optimization, this loss function "fines" the ANN if the distance measure between similar-sounding music is greater than the distance between differing music. The additional term g is a hyper-parameter that allows us to adjust the distance between the vectors of similar and differing music. The similar music was defined as tracks having more than 10 identical tags with the target song; as differing music, we chose a track from another genre that had no common tags with the target song.

$$L(p_i, p_i^+, p_i^-) = max\left\{0, g + D\left(f(p_i), f(p_i^+)\right) - D\left(f(p_i), f(p_i^-)\right)\right\} \quad (3)$$

The authors of [8] simultaneously trained 3 neural networks to process triplets. Due to the technical restrictions of GPU memory size used to train the neural network, we used just one ANN instead of three. We chose a conventional convolutional neural network consisting of four convolutional layers with max-pooling. ReLU was used as the activation function. Tensor normalization according to the number of channels was also applied. The ANN architecture is shown in Fig. 3. Increasing convolutional layers and residual connections has not considerably improved the results of the experiments.

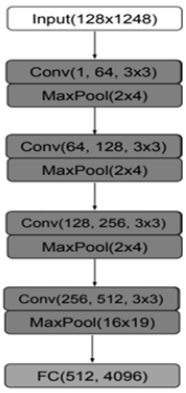

Fig. 3. Artificial neural network architecture

The triplet spectrograms were consecutively fed to the neural network. The spectrogram dimensionality is 128x1248, and the neural network gives us a vector of the dimensionality 1x4096. The error was calculated based on the obtained vector representations, and then the network parameters were recalculated. Music compositions having the same tags were considered similar, while compositions from different genres and lacking common tags were considered different. The Euclidean distance was used as the distance metrics between the vectors.

In general, the process of recommendations building using the ANN can be represented as follows:

- Obtain mel-frequency spectrograms for audio recordings;
- Train the ANN to display spectrogram as a vector representation in such a way, that vectors of similar-sounding songs are closer to each other.
- Using a trained network, map each song to a vector representation;
- In the resulting vector space, calculate the distance measure between the vectors;
- For each musical composition from the test set, create a list of 10 nearest neighbors in the vector space.

The quality metrics of the received recommendations are presented in Table II.

VI. DATASET

Within the framework of a recommender system, it is important to understand that we are primarily interested in whether the system correctly predicts that the user follows our recommendations, that is, it is necessary to evaluate the "usefulness" of our recommendations to the user. In order to do this, we need to analyze the history of the user's interaction with the system objects. However, by now there are no datasets in the public domain that contain audio files and user listening history.

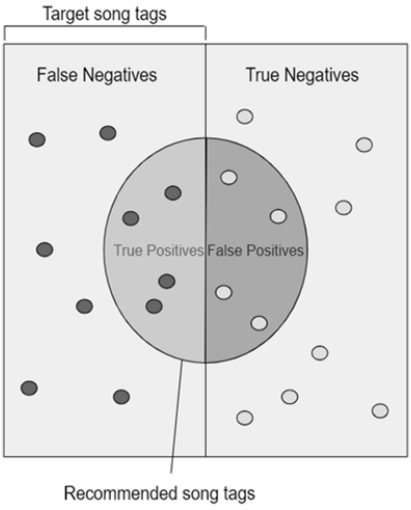

Fig. 4. Tag splitting into relevant and irrelevant ones

To build a model of a recommender system and to find relevant features, we analyzed several music datasets with open access. The most popular music dataset is the Million songs dataset (MSD) [9]. According to copyright, the MSD contains only pre-calculated acoustic characteristics of the songs and no audio recordings. Considering that our task is to determine the music similarity based on the sound of music, we required to analyze audio files themselves. For this reason, we chose an alternative dataset, namely, the Free Music Archive (FMA) [10]. The dataset was created in 2017. In the full version, it contains meta data and full recordings of 106,000 tracks. At this stage of the experiment, a shortened version of this dataset was used, containing 8000 track fragments with the duration of 30 seconds each. We believe, that considering the modern music it is enough to use a 30-second fragment instead of the full recording to evaluate the similarity of two compositions. Besides, this approach significantly reduces the amount of information processed. If it is technically possible, the use of full audio recordings would improve the quality of recommendations. FMA provides no free access to the history of user's listening to music tracks.

VII. QUALITY EVALUATION OF THE RECOMMENDER SYSTEM

Quality evaluation of an algorithm is one of the most important steps in machine learning tasks. The correct task setting, and the subsequent choice of the quality evaluation metrics greatly determine the result of all the work done. Two approaches are commonly used to evaluate the quality of a content-based recommender system: an objective approach and a subjective one [11]. During the objective evaluation, the quality is measured using some quantitative indicators, i.e., metrics. During the subjective evaluation, the quality is measured depending on the ratings of a relatively small group of people who imitate the target audience of the service.

In the current research we used only an objective evaluation of the developed recommender system. To determine the similarity of the music sounding, we considered the matching of the external characteristics (such as genre, description, tags, etc.) of the target track and the recommended one. In this study it was decided to use tags as the most relevant description of music.

The tag is an identifier for data categorization, description and retrieval. In music services the tag is a keyword or phrase that describes music. We have obtained a licensed access to API of Last.FM, the world's largest online music service. Using API, the tags of 8000 songs were downloaded, and if the service provided no information about the track itself, then the tags specifying the artist were used instead. These tags were assigned by users and they describe genre, music and performance styles, mood and emotions.

Then, a test sample of 100 songs was randomly chosen from the entire set of available musical compositions and this set was fixed. The only condition used to choose the tracks for the test sample was having at least 10 tags for each track. All other tracks were moved to the training sample.

Since tracks had varying number of tags, a direct calculation of the number of matching tags for the target track and the recommended one could not be considered a reliable estimator of the recommendation quality. Thus, this estimator must be normalized. We can consider the situation from the

point of view of tags classification into relevant and irrelevant ones, as Fig. 4 shows.

Such tag splitting allows us to use classical metrics that are

$$precision = \frac{TP}{TP + FP} \quad (4)$$

$$recall = \frac{TP}{TP + FN} \quad (5)$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \quad (6)$$

$$precision@10 = \frac{\sum r^{F1 \geq 0.4}}{10} \quad (7)$$

often used in machine learning tasks (4, 5):

The true positives (TP) are the cases when the tags of the target and the recommended song match. The false positives (FP) are the cases when the tags of the recommended track are irrelevant, the false negatives (FN) are "unrecognized" tags of the target track.

However, if the target composition has a small number of tags and the recommended composition has many tags, even a small number of matches will greatly overestimate the recall. The situation with precision is just the opposite. To obtain the most reliable quality evaluation of the recommendation, it is necessary to combine precision and recall. We took the F1 score (6) as a quantitative estimator of the relevance of the recommended track to the target one. A threshold value of 0.4 was chosen: if the F1 score is greater or equal to the threshold value, the recommendation is considered relevant to the target track, otherwise it is considered irrelevant (7). The threshold was chosen based on the authors' subjective evaluation of the music similarity. Thus, for each song from the test sample a set of 10 tracks from the training sample was recommended. Summing up the total number of relevant tracks and averaging it by the number of recommendations (equal to 10) and the test sample size (equal to 100), we obtain a single objective quality evaluation of the recommender system. A similar way of measuring quality was described in [12].

It is also necessary to correctly rank the list of recommendations according to their level of relevance to the user. To evaluate the quality of objects ranking, the nDCG measure (normalized discounted cumulative gain) was used. This metric not only evaluates whether the recommended object was relevant, but also takes into account the order of the object in the list of recommendations, as it is important that the most relevant objects are at the top of the recommendation list. Discounting means that objects placed at the top of the list are especially important, while the importance decreases towards the end of the list. In the current research, similar sounding tracks were considered to be relevant objects.

VIII. RESULTS

Table II provides the results of the recommendation quality based on the methods described in this paper. The best results, both from the point of view of recommendations relevance and recommendations ranking, showed the model that used the ANN trained on triplets.

The relatively low values of quality metrics for all implemented models can be explained by the influence of the following factors:

- Limited data. We were searching for similar sounding music among 8000 tracks only, although some modern music services include millions of songs.
- Nonuniform distribution of the tags number among the tracks. As a result, the recommended song may be similar to the target track, and the recommender system will rank it highly; but in case of insufficient number of tags, this recommendation will not be considered relevant when calculating the quality evaluation metrics.

TABLE II. RECOMMENDATION QUALITY METRICS

Recommender system	mean precision@10	mean nDCG
Random recommendation	0.006	0.006
Genre-specific random recommendations	0.066	0.066
Acoustic characteristics analysis	0.112	0.125
Artificial neural network	0.148	0.164

As we can see, the common approach significantly outcomes the random model both in finding the relevant recommendations and in results ranking. Fig. 5 shows the precision@10 metrics distribution obtained during the testing of the reference model and the genre-specific random recommendations model. The difference in results is statistically significant as it was proved by the Student and Mann-Whitney criteria.

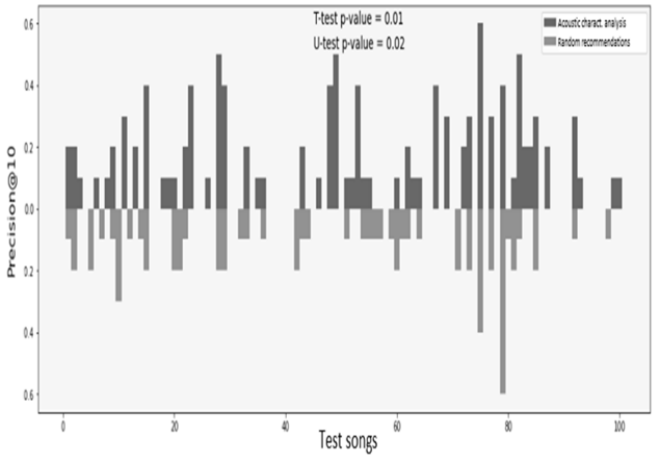

Fig. 5. Presicion@10 metric distribution: acoustic features analysis (top), random recommendations (bottom)

Fig. 6 shows the distributions of the precision@10 metrics, displaying the recommendation accuracy obtained for test

audio recordings using two methods: the ANN model and the more traditional method using acoustic characteristics extraction. The difference in results is statistically significant as proved by the Mann-Whitney test.

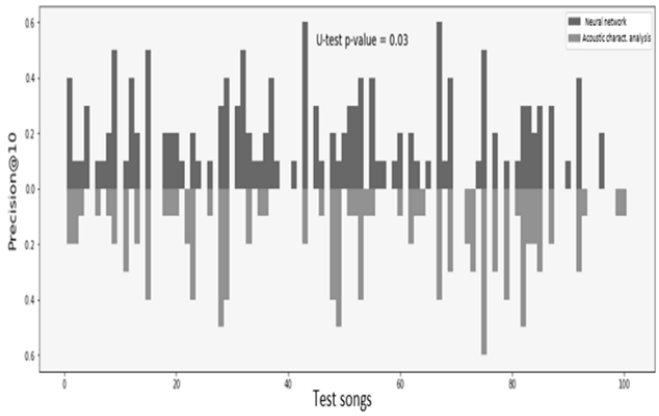

Fig. 6. The precision@10 metrics distribution: neural networks (top), acoustic features analysis (bottom)

To compare the results of the approaches used, we mapped two vector spaces on the plane (Fig. 7). To better evaluate the quality of songs clustering according to the genre, only the most distinguishable music genres are shown in the picture, such as hip-hop, rock, electronic, folk, and instrumental music (Fig. 8). Visually, the quality of genre specific clustering of vector representations is higher when a neural network is used.

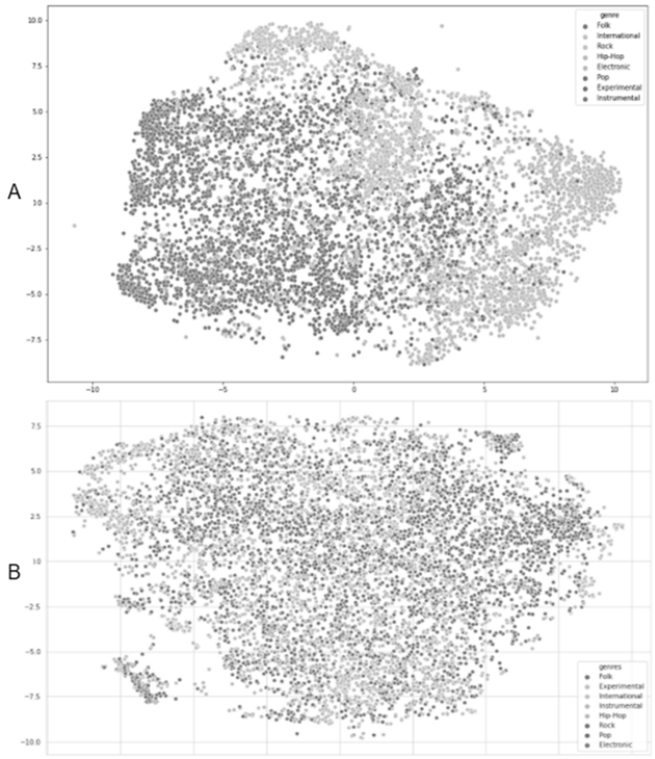

Fig. 7. Vector representations mapping. A – using neural network, B – using acoustic characteristics

Since the assignment to the musical genre is closely related to the sound of a musical composition, we can conclude that the vector representations obtained by the ANN contain more information about the music. This confirms our assumptions.

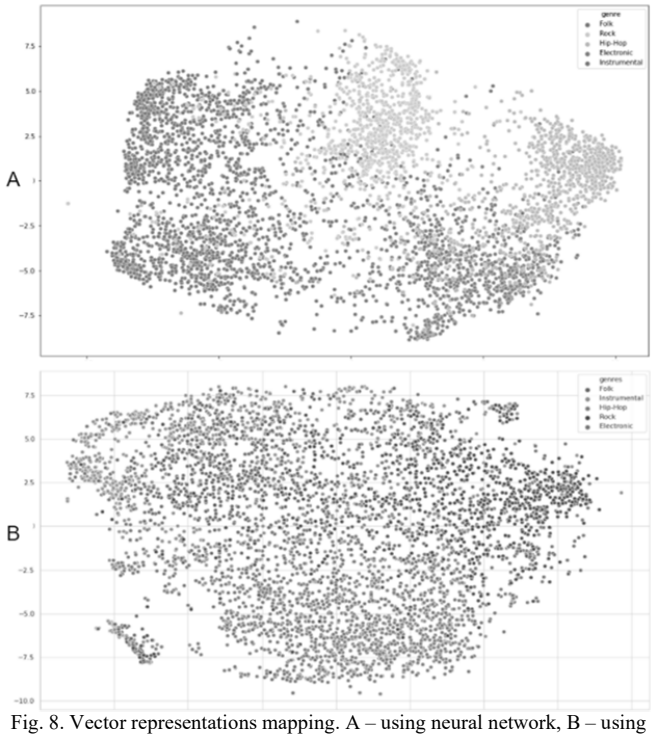

Fig. 8. Vector representations mapping. A – using neural network, B – using acoustic characteristics.

IX. CONCLUSIONS

In our research, we built a model of a content-based music recommender system based solely on the music characteristics. We investigated a method based on the extraction and further analysis of acoustic characteristics of the audio signals. The results significantly outperformed random recommendations. We also managed to improve the quality of recommendations by using the ANN.

REFERENCES

[1] P. Knees and M. Schedl, *Music similarity and retrieval: an introduction to audio- and web-based strategies.* Berlin: Springer, 2016.
[2] A. van den Oord, S. Dieleman and B. Schrauwen, “Deep content-based music recommendation”. 2013.
[3] R. Lu, K. Wu, Z. Duan and C. Zhang, “Deep ranking: triplet MatchNet for music learning”. 2017.
[4] J. Cleveland, D. Cheng, M. Zhou, T. Joachims and D. Turnbull, “Content-based music similarity with triplet networks”. 2020.
[5] D. Bogdanov et al. “ESSENTIA: An audio analysis library for music information retrieval”. 2013.
[6] G. Shani and A. Gunawardana, “Evaluating recommendation systems”. 2011.
[7] K. Choi, “A Tutorial on Deep Learning for Music Information Retrieval”. 2018.
[8] J. Wang et al. “Learning Fine-grained Image Similarity with Deep Ranking”. 2014.
[9] T. Bertin-Mahieux, D. P. W. Ellis, B. Whitman and P. Lamere, “The million song dataset”. 2011.
[10] M. Defferrard, K. Benzi, P. Vandergheynst and X. Bresson, “FMA: a dataset for music analysis”. 2017.
[11] K. Gurjar and Y. Moon, “A comparative analysis of music similarity measures in MIR systems”. 2018.
[12] J. Kaitila, “A content-based music recommender system”. 2017.