State of art: Music Recommendation Systems and Music Processing

Tanel Peet

March 6, 2016

Contents

1 Music Recommendation Systems

Creating music recommendations for users is a task which involves cooperation from several fields, such as psychology, signal processing, computer science and machine learning, to name a few. The amounts of available data for analysis and success in recommendation systems, have made Music Recommendation Systems popular topic of research for computer scientists all over the world. The recommendation of a song is dependent on many variables, such as one's mood, activity or weather conditions. Large amounts of data and complexity of the problem, makes this task appropriate for Machine Learning algorithms [4].

There are many approaches to music recommendation and it is important to distinguish music recommendations from music similarity. Musical similarity does not not mean that the similar song might also be a good recommendation, because in addition to song itself, various other things are important, such as cultural background of the artist and listener. Music similarity is usually based on comparing the audio signal of different songs and measuring their similarity. Music recommendation is much more complex problem and could use music similarity as one of its input features [4].

Many recommendation systems use patterns, generated by listeners, to provide good recommendations. Amazon and Netflix are well known examples of companies who own big part of their success to this kind of recommendation systems. This approach is called *collaborative filtering* and it is generally thought that it renders better results than using the data about the song not its usage. The latter approach is called content based filtering. The problem with collaborative filtering is that it does not perform well on examples which has no usage data. This is called the *cold start problem* and it is problematic for songs with no usage data. Content based filtering, which gets its data by analyzing audio signal and using its metadata, does not suffer from this problem, but does not perform so good otherwise[26]. In the following sections we look into different information sources used in music information retrieval and explain different strategies for extracting this information.

1.1 Information Sources

The amounts of data available for researchers is rapidly increasing. Advancements in data science are creating new ways of gathering that large amount of data and analyzing it. Music recommendation systems can use a variety of different data sources which are described later in this section. These sources can be divided into *metadata* and *audio content*. From Table 1 it could be seen that all the music recommendation services viewed in this paper use metadata and only some are in addition using audio content. It is shown in several research papers that best results are achieved when these two approaches are used together [4].

1.1.1 Metadata

Musical metadata is information about music which is not extracted from the audio signal. Traditionally it contained the title of the song, artist name, album name, music genre, track number, year of release etc. This kind of information is often included in .mp3 files as ID3 tags. Thanks to advances in data science, the collected metadata includes much more, for example properties like danceability, instrumentation or mood. Metadata sources might include manual expert annotations, annotations mined from the Internet or collaborative filtering data generated by the listener. Collaborative filtering data is probably the most established types of metadata, because it can be applied to virtually any domain of recommendations [4].

1.1.2 Audio Content

Audio content is the information which is extracted from raw audio signal. When comparing semantic data, the audio content can be divided into *low-level* and *high-level*. Low-level content can include *timbral*, *tonal* or *temporal* information. Temporal information might be dynamics, rhythmic properties, musical structure, evolution of loudness or timbral characteristics. Low-level information can help addressing different aspects of music and can be used for recommendation algorithms, but it has only been started to use recently thanks to the advancements of music analysis tools. Timbre is the mostly used information source in the studies in this area, while other acoustical and musical aspects are often ignored [4].

Examples of high-level information are automatic genre classification, auto-tagging by genre, automatically classified mood or instrumentation by audio content. High-level information is rarely employed for music recommendations, although there is some evidence, that it has some advantages [4].

1.2 Information elicitation strategies

Gathering information about songs is one of the hardest tasks in creating music recommendation systems and there are several ways to approach this problem. There are several services which offer music recommendations and have put a lot of effort in collecting the information about songs. Table 1 shows some of the commercial music recommendation services and their methods of collecting data.

Service	**Source of data**
Pandora	Musicologists take surveys
Songza	Editors or music fans make play lists
Last.fm	Activity data, tags on artists and songs, acoustic analysis
All music guide	Music editors & writers
Amazon	Purchase & browsing history
iTunes Genius	Purchase data, activity data from iTunes
Echo Nest	Acoustic analysis, text analysis

Table 1: Music recommendation services and what data sources they are using[27]

The manual data collection by musicologists done by Pandora gets a lot of noise free information about each song but does not scale and needs a lot of human resources. The users get great suggestions, but after some time, the recommendations start to repeat themselves and new songs do not reach the system fast enough. Gathering information automatically is much faster and scales well, but the data about songs are more noisy and many properties might be labeled incorrectly [27].

The best results are achieved when data is gathered from the listener. The sources of knowledge about the listener's preferences can be divided into two - *implicit* and *explicit*. Both have their own features and problems, but best results are achieved then these two strategies are combined[4].

1.2.1 Implicit feedback

Implicit information is the data obtained from the usage of the system by the listener and measuring the interaction with different songs. The advantage of implicit feedback is that it does not need any extra effort from the user. The negative aspect of it is that information tends to be more noisy. Examples of implicit feedback: [4]

- listening behavior
  - top tracks list
  - top artists list
  - recently listened artists/tracks lists
- consumption history
  - track/album/artist purchases
- tagging history
  - user tags
  - tracks/albums/artist tagged by the user
- activity in social networks
  - sharing activity
  - commenting activity
- the user's music collection

1.2.2 Explicit feedback

Explicit information is obtained by directly querying the listener. The problem of explicit is that users are not eager to provide ratings or fill up surveys. [4]. Examples of explicit feedback:

- user surveying for preference examples
  - artist lists
  - track lists
  - recording label lists
  - favorite genres/styles
  - favorite moods, situations, locations for music
  - musical properties
    - melody, rhythm, timbre, instruments, expressiveness
    - voice, lyrics
- user feedback
  - love/skip buttons
  - user ratings

1.3 Conclusion

Music Recommendation System need to gather a lot of information from different sources in order to offer good recommendations. The best result are yielded when combination of data from audio signal and metadata is used. It is also important to use feedback from users, either implicit or if possible, explicit. Listener's choice of music can be largely affected by one's mood, activity or weather. That is why only comparing audio signals is not enough when trying to give recommendations. [4]. In addition to examples of data gathered in this paper, future developments are starting to use data from users mobile device. Such data might use data from sensors, texts or tweets to predict the mood of the user and give context based recommendations.

2 Music processing

With the wide use of computers and digital media, there are big quantities of data generated. As a result, a large number of Machine Learning (ML) algorithms have been developed to help navigate and make sense of the data. Music Information Retrieval (MIR) is one of these disciplines, which in addition to machine learning is drawing from several different fields, including electrical engineering, music psychology, computer science and on-line social interactions. The term MIR could be defined as the study of information related to musical activity and it has many different applications, including melody extraction, chord recognition, beat tracking, tempo estimation, instrument identification, music similarity genre classification and mood prediction [11].

The area of music information retrieval has become more and more popular with each year. Compared to the beginning years of MIR research, the number of publications has gone up more than tenfold. Although the research in the field is getting more popular, the progress has become decelerating if not altogether stalled in recent years. Among a broad spectrum of applications the vast majority of music signal processing systems use a two-stage process of feature extraction and classification [24, 11].

2.1 Hand-crafted features

Until recent years, the features have usually been handcrafted by experts who have good domain knowledge and deep understanding of signal processing [12]. The features could be separated to groups in different ways, one of which is labeling them to *high-level*, *mid-level* and *low-level* features. Examples of high level features are genre, mood and instrument detection. Mid-level features include rhythm, pitch, harmony. Low-level features could be in turn divided into two classes - *timbre* and *temporal* [9]. In this paper we are concentrating on low-level features because all others are build upon these.

Timbre features are capturing the tonal quality of sound, usually in frames with 10-100 ms duration. Even though timbre is the most popular low-level musical feature used in MIR, it still has number of issues. The frame length used in extracting a feature is usually fixed size and does not account for musical events. In addition phase information is usually discarded, although it might contain useful information [9].

Temporal features are usually constructed on top of the timbre features and thus, they are capturing the variation and evolution of timbre over time. The simplest type of temporal features are different statistical moments, like mean, variance, covariance. The issue with temporal features is that they need more computing power. [9]

Low-level feature classes can be in turn divided to features. Table 2 shows low-level features commonly used in MIR applications. Different features are crafted to solve different problems, but the most popular low-level feature is Mel-frequency Cepstrum Coefficients.

Musical properties	**Feature type**
Timbre	Zero Crossing Rate (ZCR) Spectral Centroid (SC) Spectral Roll-off (SR) Spectral Flux (SF) Spectral Bandwidth (SB) Spectral Flatness Measure (SFM) Spectral Crest Factor (SCF) Amplitude Spectrum Envelop (ASE) Octave based Spectral Contrast (OSC) Daubechies Wavelet Coef Histogram (DWCH) Mel-frequency Cepstrum Coefficient (MFCC) Fourier Cepstrum Coefficient (FCC) Linear Predictive Cepstrum Coefficient (LPCC) Stereo Panning Spectrum Features (SPSF)
Temporal	Statistical Moments (SM) Amplitude Modulation (AM) Auto-Regressive Modeling (ARM)

Table 2: Commonly used low-level features in MIR [9]

Mel-frequency Cepstrum Coefficients (MFCC) have become the standard low-level feature to use in most of the MIR problems [12]. MFCC was originally created for solving speech recognition problems, but was adapted for retrieving information from music [6, 20]. MFCC feature extraction from audio signal starts by taking Fourier transform of a windowed excerpt of a signal. The powers of the results are then mapped onto the *Mel scale* using triangular overlapping window. After taking logs of each of the Mel frequencies, the *discrete cosine transform* is taken of the Mel log powers. The amplitudes of the resulting spectrum are the MFCCs [21].

2.2 Automatic Feature Learning

In recent years the MIR community has starting to realize that hand-crafted features might be the bottleneck, which is the reason of the decelerating progress. Until recently, the progress was achieved by manually creating better features and implementing more complex classification algorithms, but this approach is now hitting the ceiling. The problem of finding the optimal features and finding the best classification algorithm, has a large solution space. Finding good solutions in that vast space is time consuming and difficult task. Moreover, the solution might be good for only the problem it was crafted for. The manually developed feature extraction method might also be too *shallow*, containing only few non-linear transformations, although sounds lives on a highly non-linear manifold. Short-time signal analysis is also problematic, because majority of musical experiences is joined in parts which are rather measured in seconds, than milliseconds. [11]

The solution to previously described problems is believed to be found in using *neural network* algorithms to achieve automatic feature learning. Neural networks are successfully proven itself in various other domains, but has not been found wide usage in music informatics yet [11]. Multiple researches have shown better results using automatic feature learning, instead of using the most successful hand-crafted features [16, 17, 23]. The techniques used in successful automatic feature learning researches are derived from other fields of research, like image and natural language processing [18].

2.2.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are multi-layer architectures where successive layers learn progressively higher-level features. The layers are trained simultaneously to minimize the overall loss function. CNN is usually composed of a *feature detection layer*, followed by *feature pooling layer* [1]. CNNs are typically used for image processing which have similar characteristic to audio signals, although the locality is temporal rather than spatial [7]. Several researches [16, 17, 7, 26] show better results using convolutional networks for feature extraction, compared to using hand-crafted features.

Convolutional network is used by Li et. al. in [17]. The GTZAN dataset, described in detail in section 3, was used. Each 30 second audio clip was segmented in a way that MFCC results featured 190 frames. The input to CNN were feature maps which contained 13 MFCC feature vectors with a length of 190 frames. The CNN consisted of three convolutional layers and the output layer. Different generic classification methods, such as Decision Trees, Support Vector Machines etc., were used on the outputs of the last CNN layer and the results were aggregated in a majority voting process to produce correct labels. The experiments of the research concluded that the model that was used did not generalize the training results to unseen musical data. Although the accuracy was not good, the research showed a way to use highly scalable CNNs in the field of MIR and suggested improvements for future work.

Traditional bag-of-words representation of audio signals is compared to CNNs in automatic music recommendation task by Oord et. al [26]. For bag-of-words the MFCCs were used as a input and a multilayer perceptron was used for obtaining predictions. For CNNs, an intermediate time-frequency representation of the audio was used as input. The paper does not describe in detail the parameters used in network, although it mentions that the structure consisted of alternating feature extraction layers and pooling layers. The dropout regularization was also experimented, but it did not yield any significant improvements. The Million Song Dataset [2] was used in the research to show that CNNs can outperform traditionally used bag-of-words representation.

Schluter and Bock study musical onset detections with the help of CNNs in their research [22]. They use a custom dataset of about 102 minutes of monophonic and polyphonic instrumental recordings and popular music excerpts annotated with 25,927 onsets. The input has a 3 channels which contain Mel-scaled magnitude spectrograms with different window sizes. The input was fed to CNN which contains two convolutional layers, which both were followed by max-pool layers. After the last max-pool layer, came two fully connected layers, which resulted in a single output unit predicting onsets. The F-score of the CNN was about 1% point better than the state-of-art approach before the research. The results were improved more by the usage of Dropout, Fuzziness and Rectified Linear units.

2.2.2 Deep Belief Networks

The deep Belief Network (DBN) is a neural network which is constructed from several layers of Restricted Boltzmann Machines (RBMs). Top-level features learned by DBNs can be used to train discriminative models which have been applied successfully to image processing problems. Recently DBNs have also started to be used for feature extraction in music information retrieval tasks.

Lee et. al. looked into using convolutional networks combined with deep belief networks on multiple applications in speech recognition and music information retrieval tasks in [16]. They compared MFCCs to the approach of extracting feature with convolutional deep belief networks (CDBN) on speaker identification, speaker gender classification, phone classification, music genre classification and music artist classification. In most cases the features from CDBNs outperformed MFCCs and in a few cases when the performance was similar, it was shown that the linear combination of MFCCs and CDBNs features yielded the best results. The research used TIMID [8] dataset for speech recognition tasks and ISMIR[1] dataset for music recognition tasks.

Deep Belief Networks were also used by Hamel and Eck in [10]. They used GTZAN dataset, segmenting the audio into 1024 sample frames which corresponds to 46.6 ms of music. For each frame, the Discrete Fourier Transform (DFT) was calculated. The DBNs were pre-trained with the training set in an unsupervised manner and later fine-tuned on the same training set. Validation data was used to do early-stopping. SVMs were used for classification tasks and it was shown that their method performed better than using MFCCs as features. Additionally features were aggregated with a 5 second window, which drastically increased the performance, both in accuracy and also in classification speed.

2.2.3 Other Deep Learning Algorithms

In addition to Convolutional Neural Networks and Deep Belief Networks, Recurrent Neural Networks (RNN) are gaining popularity. RNNs incorporate an internal memory which makes it well suited to model temporal sequences, such as different properties of music. Several research groups have used RNNs in their research with good results. Bock et. al. used RNNs with audio signal transformed by Short Time Fourier Transform to frequency domain for on-line real-time onset detection[3]. Marchi et. al. used also RNNs, although different approach was implemented, for audio onset detection [19]. Boulanger-Lewandowski et. al. successfully used RNNs integrated in the Non-Negative Matrix Factorization (NMF) framework for audio source separation applications.

An alternative to Deep Belief Networks are Auto-encoders. Although Auto-encoders have not had big success yet in the field of automatic feature extraction, it has shown promising results. Klec and Korzinek wrote two papers where they used Sparse Auto-encoders and Scattering Wavelet Transform as input for music genre recognition [15, 14] on GTZAN dataset. The latter article used nearly 10,000 additional musical tracks from Jamendo music-sharing platform for training. Deep auto-encoders were proposed by Jang et. al. in their paper [13] for audio source separation.

[1] Available from http://ismir2004.ismir.net/ISMIR_Contest.html

2.3 Classification

After the feature extraction the features have to be classified. There are several algorithms for classification, including Naive Bayes, Logistic Regression, Support Vector Machines (SVMs), k- Nearest Neighbors (KNN), Decision Trees (DTs) and Neural Networks (NNs). The algorithm selection can be dependent on features used as input to classifiers. The examples in this section show that different kind of classification algorithms were successfully used for multiple tasks in Music Information Retrieval. The studied papers all used different approach to classification and were concentrated on automatic feature learning, instead of classification.

WEKA [2] machine learning system was used by Li. et. al. for classification tasks using various different models, showing best results using tree classifiers such as J48, Attribute Selected Classifier etc [17]. Hamel and Eck used non-linear Support Vector Machines on features learned by Deep Belief Networks for genre recognition and auto-tagging [10]. In addition, they successfully applied aggregation of frame-level features which yielded better results and faster training of the SVMs. Windowed Logistic Regression (WLR) and Naive Bayes (NB) were used for key detection, artist and genre recognition using pretrained Convolutional Deep Belief Network, where WLR outperformed NB in every task [7]. Multiple Linear Regression was used as a output of the DBN by Schmidt and Kim in [23] for learning emotion-based acoustic features.

2.4 Conclusion

The examples of using different Deep Learning algorithms for automatic feature extraction have shown good results in several fields of Music Information Retrieval. In most cases they outperform the results of hand-crafted features and make the classification process easier and faster. The large amount of various deep learning methods and classifiers used in researches show that MIR community has not yet found consensus on the best algorithms to use on MIR tasks. There is still a lot research to be done and many ways to improve, even on the simplest of MIR tasks.

3 Datasets

This section offers a quick review of some of the most used dataset in Music Information Retrieval tasks.

The Million Song Dataset (MSD) [2] was created in order to make a large dataset available to researchers. The usage of songs for research has been problematic because of the licensing issues of the songs. MSD by-passed these problems by providing the songs without the audio signal. Instead the main acoustic features, like *pitches*, *timbre* and *loudness*, are offered. In addition, a lot of metadata is available for each song. As the name states, the dataset contains million songs which makes 280 GB of data.

[2] Available from http://www.cs.waikato.ac.nz/ml/weka/index.html

GTZAN Genre Collection [25] was created for the purpose of genre recognition tasks. It contains 10 genres, each represented by 100 30 second tracks. The track have sampling rate of 22050 Hz and they are Mono 16-bit audio files in .wav format. In order to represent a variety of recording conditions, the data was gathered from various places including CDs, radio and microphone recordings. Although, this dataset is widely used in MIR community, one has to be careful, because the dataset has some faults to be aware of, including repetitions, mislabellings and distortions [24].

IRMAS [5] dataset includes musical excerpts with annotations of the predominant instrument(s) present. It is intended to be used for recognition of predominant instruments in musical audio. There are 11 different instrument classes with over 6000 audio files in 16 bit stereo .wav format, sampled at 44.1 kHz. The excerpt length is 3 seconds and they are obtained from over 2000 distinct recordings. The data is not balanced.

References

[1] Yoshua Bengio, Yann LeCun, et al. Scaling learning algorithms towards ai. *Large-scale kernel machines*, 34(5), 2007.

[2] Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, 2011.

[3] Sebastian Böck, Andreas Arzt, Florian Krebs, and Markus Schedl. Online real-time onset detection with recurrent neural networks. In *Proceedings of the 15th International Conference on Digital Audio Effects (DAFx-12), York, UK*, 2012.

[4] Dmitry Bogdanov et al. From music similarity to music recommendation: Computational approaches based on audio features and metadata. 2013.

[5] J. Bosch, J. Janer, Ferdinand Fuhrmann, and P. Herrera. A comparison of sound segregation techniques for predominant instrument recognition in musical audio signals. In *13th International Society for Music Information Retrieval Conference (ISMIR 2012)*, pages 559–564, Porto, Portugal, 08/10/2012 2012.

[6] JS Bridle and MD Brown. An experimental automatic word recognition system. *JSRU Report*, 1003(5), 1974.

[7] Sander Dieleman, Philémon Brakel, and Benjamin Schrauwen. Audio-based music classification with a pretrained convolutional network. In Anssi Klapuri and Colby Leider, editors, *Proceedings of the 12th international society for music information retrieval conference : Proc. ISMIR 2011*, pages 669–674. University of Miami, 2011.

[8] William M. Fisher, George R. Doddington, and Kathleen M. Goudie-Marshall. The darpa speech recognition research database: Specifications and status. In *Proceedings of DARPA Workshop on Speech Recognition*, pages 93–99, 1986.

[9] Zhouyu Fu, Guojun Lu, Kai Ming Ting, and Dengsheng Zhang. A survey of audio-based music classification and annotation. *IEEE Transactions on Multimedia*, 13(2):303–319, April 2011.

[10] Philippe Hamel and Douglas Eck. Learning features from music audio with deep belief networks. In *ISMIR*, pages 339–344. Utrecht, The Netherlands, 2010.

[11] Eric J. Humphrey, Juan P. Bello, and Yann LeCun. Feature learning and deep architectures: new directions for music informatics. *Journal of Intelligent Information Systems*, 41(3):461–481, 2013.

[12] Eric J Humphrey, Juan Pablo Bello, and Yann LeCun. Moving beyond feature design: Deep architectures and automatic feature learning in music informatics. In *ISMIR*, pages 403–408. Citeseer, 2012.

[13] Gil-Jin Jang, Han-Gyu Kim, and Yung-Hwan Oh. Audio source separation using a deep autoencoder. *CoRR*, abs/1412.7193, 2014.

[14] Mariusz Kleć and Danijel Koržinek. Pre-trained deep neural network using sparse autoencoders and scattering wavelet transform for musical genre recognition. *Computer Science*, 16(2):133–144, 2015.

[15] Mariusz Kle and Danijel Korinek. Unsupervised feature pre-training of the scattering wavelet transform for musical genre recognition. *Procedia Technology*, 18:133 – 139, 2014. International workshop on Innovations in Information and Communication Science and Technology, {IICST} 2014, 3-5 September 2014, Warsaw, Poland.

[16] Honglak Lee, Peter Pham, Yan Largman, and Andrew Y. Ng. Unsupervised feature learning for audio classification using convolutional deep belief networks. In Y. Bengio, D. Schuurmans, J. D. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 1096–1104. Curran Associates, Inc., 2009.

[17] Tom LH Li, Antoni B Chan, and A Chun. Automatic musical pattern feature extraction using convolutional neural network. In *Proc. Int. Conf. Data Mining and Applications*, 2010.

[18] Martin Lngkvist, Lars Karlsson, and Amy Loutfi. A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern Recognition Letters*, 42:11 – 24, 2014.

[19] E. Marchi, G. Ferroni, F. Eyben, S. Squartini, and B. Schuller. Audio onset detection: A wavelet packet based approach with recurrent neural networks. In *Neural Networks (IJCNN), 2014 International Joint Conference on*, pages 3585–3591, July 2014.

[20] Paul Mermelstein. Distance measures for speech recognition, psychological and instrumental. *Pattern recognition and artificial intelligence*, 116:374–388, 1976.

[21] Md. Sahidullah and Goutam Saha. Design, analysis and experimental evaluation of block based transformation in {MFCC} computation for speaker recognition. *Speech Communication*, 54(4):543 – 565, 2012.

[22] J. Schluter and S. Bock. Improved musical onset detection with convolutional neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 6979–6983, May 2014.

[23] Erik M. Schmidt and Youngmoo E. Kim. Learning emotion-based acoustic features with deep belief networks. In *Applications of Signal Processing to Audio and Acoustics (WASPAA), 2011 IEEE Workshop on*, pages 65–68, Oct 2011.

[24] Bob L. Sturm. The state of the art ten years after a state of the art: Future research in music information retrieval. *Journal of New Music Research*, 43(2):147–172, 2014.

[25] George Tzanetakis and Perry Cook. Musical genre classification of audio signals. *Speech and Audio Processing, IEEE transactions on*, 10(5):293–302, 2002.

[26] Aaron van den Oord, Sander Dieleman, and Benjamin Schrauwen. Deep content-based music recommendation. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2643–2651. Curran Associates, Inc., 2013.

[27] Brian Whitman. How music recommendation works and doesnt work, 2013.