# Unsupervised Learning Approach for Identifying Sub-genres in Music Scores<sup>\*</sup>

Girija Shingte and Mathieu d'Aquin<sup>[0000-0001-7276-4702]</sup>

National University of Ireland Galway G.SHINGTE1@nuigalway.ie mathieu.daquin@nuigalway.ie

Abstract. Detecting the genre of a piece of music and whether two pieces of music are similar are subjective matters since audiences may perceive the same music differently. While the problem of the automatic detection of music genres has been studied extensively, it is still an open problem, especially when looking at sub-genres of traditional music. It can however be useful for example to discover similarities between multiple collections, to study whether a particular genre has resemblances with other genres, and to trace the origin and evolution of a particular genre. In this paper, we focus on traditional Irish music and the features and algorithms that can be used for analyzing such music through structured data (music scores). More precisely, audio, spectral, and statistical features of music scores are extracted to be used as input to unsupervised clustering methods to better understand how those features and methods can help identifying sub-genres in a music collection, and support "genre-driven" similarity-based retrieval of music in such a collection. We in particular show which features best support such tasks, and how a slight modification of the K-Means algorithm to introduce feature weights achieves good performance. We also discuss the possible use of those results, especially through a demonstration application for music information retrieval in Irish traditional music collections.

**Keywords:** Music Score · Music Classification · Music Information Retrieval · Music Similarity.

# 1 Introduction

Music is an integral part of all cultures across the world and greatly influences an individual's emotions, productivity, and behavior. In today's digitalized era, on-demand access to all kinds of music has been significantly simplified and there has been extensive research to enhance user experience. The discipline that studies techniques for the retrieval of information from music-related data is Music

<sup>\*</sup> This work has been partly funded by Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289\_P2, Insight SFI Research Centre for Data Analytics. The work was carried out as a project of the Masters Programme on Data Analytics at NUI Galway.

Informational Retrieval. A simple definition of Music Information Retrieval as stated by the International Society for Music Information Retrieval (ISMIR) is "processing, searching, organizing and accessing music-related data".

Music Information Retrieval as an interdisciplinary field integrates music theory, sound engineering, and machine learning. Music-related data is available in abundance due to a variety of music modalities such as audio representations, symbolic notations, and meta-data representations as well as large-scale commercialization of music on mobile and web platforms [7]. This availability and commercialization of music-data have given rise to extensive consumption of music from users thus motivating the music industry to render user-friendly services and encouraging researchers to model newer techniques in Music Information Retrieval.

However, Music Information Retrieval is not restricted to popular Western music. It also encompasses traditional or ethnic music. The discipline that studies ethnic music is Ethnomusicology. Oramas and Cornelis [9] precisely describe Ethnomusicology as "the study and understanding of music by comparing different cultures to find musical universalities and origin of music". Ethnic music is different from western music in multiple ways.

The objective of this paper is to explore features and clustering techniques for sub-genre detection in Irish traditional music, as an example of ethnic music. A genre usually describes the style of a piece of music. Given a genre and identified sub-genres, studying the similarities and dissimilarities between the sub-genres can help in understanding the correlation between the genres. For example: if a collection of Irish traditional music is compared with a collection of Scottish traditional music, the similarity between the sub-genres as well as individual tunes within these collections can be determined. This could further help determine whether the Irish music culture has resemblances with Scottish music. For example, archives of genres can be clustered to identify sub-genres and these sub-genres can be compared with newer tunes. Thus an understanding of the evolution of newer tunes can be gained by tracing its origins.

In this paper, we therefore aim to understand which music representations are more suitable to support sub-genre detection in Irish traditional music, and how clustering methods perform in sub-genre detection. We test various features extracted from music scores using a common clustering method (K-Means), and also show how an adapted version of K-Means to include weights for different feature sets (e.g. pitch, beat, spectral features) achieves better performance on a common collection of Irish traditional song scores.

# 2 Related work

Our work generally falls in the area of music information retrieval [1], which is concerned with the retrieval of information from music. Several popular applications have been developed in the past to support music information retrieval, for modern western music, as well as for traditional music. Those naturally rely on a notion of music similarity which is central to the work presented here, including:

- **Shazam:** Shazam<sup>1</sup> allows users to record audio for 10-12 seconds (irrespective of the place and presence of noise). It then checks for a perfect match on its server using the audio fingerprint of the recording. If a perfect match is found, it returns the title, artist and album of the song. There is also a provision in Shazam to redirect the user to an application/webpage where the returned music score can be played.
- **SoundHound:** SoundHound<sup>2</sup> also identifies songs based on an audio recording like Shazam and returns meta-data for the identified tune. However, additional provisions include returning live-lyrics of the tune being played and returning lists of tunes similar to the one recorded.
- **TunePal:** TunePal<sup>3</sup> is an application that allows users to identify traditional Irish tunes either by entering the title of the tune or by recording a tune. It makes use of the ABC notation<sup>4</sup> for the tunes and retrieves similar tunes using Natural Language Processing.

Various features are traditionally used to support information retrieval, including pitch, timing, etc. [1]. In this paper, we employ several more or less "intuitive" representations of those features to cluster tunes into sub-genres. In [5], a classification of the features to support Music Information Retrieval in ethnic music is provided which separate them into three groups:

- Low-level descriptors: Intrinsic properties of music scores that can be extracted with the help of sound engineering techniques, including frequency, audio spectrum, pitch, duration, beat onset and offset, etc.
- Mid-level descriptors: Features that capture what the audiences are listening, including loudness of the music score, rhythm, timbre, tonality, etc.
- **High-level descriptors:** Overall interpretation of the music score, including the mood, genre, expression or the emotion of the tune.

The features we use here fall into the first two categories, being extracted from both the music scores and the audio signal from rendering those scores computationally.

Genres are usually assigned to music scores manually, but when it comes to a large collection of data, manual allocation of genres to music scores is not possible. Both supervised and unsupervised machine learning algorithms have been used for automatic genre identification. A number of different approaches can be used to identify genres automatically, including content-based approaches (which we employ here), semantic analysis or collaborative filtering, and it can be achieved either in a supervised or unsupervised way. Below, we list some of the more prominent work in genre identification in both cases.

<sup>&</sup>lt;sup>1</sup> https://www.shazam.com/

<sup>&</sup>lt;sup>2</sup> https://www.soundhound.com/

<sup>&</sup>lt;sup>3</sup> https://tunepal.org/

<sup>&</sup>lt;sup>4</sup> http://abcnotation.com/

Using supervised methods: [11] used a hybrid approach in which spectral features, rhythmic features, and pitch content were extracted from the music scores, which were then classified using Support Vector Machine. [8] used J48, which is the Java implementation of C4.5, to classify non-western music tunes based on their genres, obtaining an average accuracy of 75%. [8] also made use of a One R Classifier in which rules are made for every predictor and the one giving the smallest total error is selected as the rule. They obtained an accuracy of around 65%. [4] used an ensemble model in which multiple classifiers were combined to improve the overall accuracy, obtaining 75-80%.

Using unsupervised methods: [3] used the K-Means clustering algorithm on a dataset that contained Classical, Rock, Jazz, Hip-Hop, and EDM music. [10], on the other hand, used Constraint-based clustering for identifying music genres. Two types of constraints were considered, positive constraints and negative constraints. The positive constraints consisted of attributes that were expected to be together based on the content-based description, whereas negative constraints consisted attributes that were not expected to be together. [6] used a self-organising map and observed that better and clearer results could be obtained compared to other clustering approaches.

In this paper, in contrast with those approaches, we focus on clustering music using features extracted from scores that are obtained from collections of traditional Irish music tunes. The objective, beyond obtaining genre-related clusters for those collections, is to better understand which aspects and features of Irish traditional music tunes are more relevant to their comparison, and how to effectively represent them.

## 3 Data collection and pre-processing

The music data used for this research was available either in the form of ABC notation files or in the form of MIDI files. Both are symbolic representation formats which represent scores rather than audio signals. While MIDI represents tunes through events (e.g. a note starts playing at a time, with a certain pitch and a certain velocity), ABC notation is a simple text format representing notes as letters and timing information through specific characters. The core information required to play the tune is common to both, so to simplify processing those, we converted all ABC notation files to MIDI.

Out of the many communities and websites that facilitate access to Irish traditional music, the data for this research was collected from the "Irish Traditional Music Archive"<sup>5</sup> and the "Session"<sup>6</sup>:

**The Session:** The Session is a community dedicated to Irish traditional music. It hosts a variety of recordings, sessions, tunes, discussions, and music

<sup>&</sup>lt;sup>5</sup> https://www.itma.ie/

<sup>&</sup>lt;sup>6</sup> https://thesession.org

collections. It also allows artists to collaborate for sessions and events. 400 ABC notation files were obtained from the Session's bulk download facility.<sup>7</sup> These files span across 4 genres: Jigs, Reels, Polkas and Barn dances. The ABC notation files were converted to MIDI files and WAV files (converted from the MIDI files through the Timidity++<sup>8</sup> software).

Irish Traditional Music Archive: The Irish Traditional Music Archive is a National Public Archive dedicated to Irish Traditional Music. The largest Irish Folk music collections can be found at ITMA. Along with song recordings, it holds information about the origin, history/evolution of tunes, metadata, information about artists, and albums, instruments, and Irish dances. ITMA also welcomes contributions from individuals/artists towards traditional Irish music. Over 6 000 Midi files from the ITMA "port" collection of digitised scores<sup>9</sup> were obtained. 32 tunes belonging to both The Session and ITMA collections were selected, with slightly different representations in the two collections. Those, in ITMA, cover 7 different genres: Jigs, Reel, Waltz, Three-Two, Slip Jigs, Hornpipes, and Barndances.

MIDI files are instructional and the messages within these files provide information such as which note is being played, its pitch, duration, velocity, loudness, etc. To read the MIDI files and extract messages from them, the python library MIDO<sup>10</sup> was used.

For extracting audio data from WAV files, the python library LibROSA<sup>11</sup> was used. LibROSA is widely used in analyzing audio, speech recognition, and sound engineering applications. It allows visualization of spectral data, spectral, temporal, and statistical feature extraction, sound filtering, onset detection, etc.

## 4 Feature engineering

For this research, multiple features were selected and tested individually and in combination. Based on the results, data were pre-processed again if needed. A summary of the feature engineering processes is diagrammatically represented in Figure 1

A brief description of the types of features considered is given below:

Audio Features: The audio features considered included the notes' duration (in tick and milliseconds, from the NOTE\_ON and NOTE\_OFF events in the MIDI files), pitch (from the note number in MIDI files) and velocity (from the corresponding field in the MIDI events), as well as the beats of the tune (extracted from the WAV files using LibROSA).

<sup>&</sup>lt;sup>7</sup> https://thesession.org/tunes/download

<sup>&</sup>lt;sup>8</sup> http://timidity.sourceforge.net/

<sup>&</sup>lt;sup>9</sup> http://port.itma.ie

<sup>&</sup>lt;sup>10</sup> https://mido.readthedocs.io/

<sup>&</sup>lt;sup>11</sup> https://librosa.github.io/

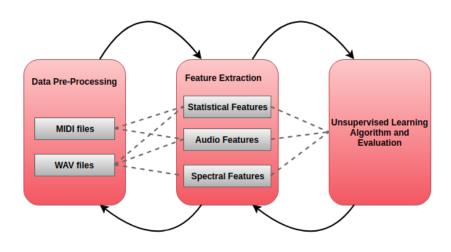


Fig. 1. Feature Engineering Process Summary.

- Mel Frequency Cepstral Coefficients (MFCCs): In simple terms, MFCCs capture the relation between a frequency of the tone the way it was perceived and actual frequency of the tone. The coefficient numbers provide the spectral energies. Lower order coefficients describe the shape of the spectrum and average energy possessed by the input signal whereas high order coefficients provide details of the sound spectrum incrementally. Twelve MFCCs for each tune were extracted from the WAV files and used as input to the unsupervised learning algorithm.
- **Statistical Features:** The statistical features considered included distributions within a given tune of the note-specific audio features, as well as of the MFCCs.

## 5 Clustering and cluster evaluation

The K-means clustering algorithm [2] is used for partitioning data into k clusters. The main objective of this algorithm is to create clusters in such a way that the points within a cluster are very close to each other (high intra-cluster cohesion), whereas the points in different clusters are away from each other (low intercluster cohesion).

In K-means clustering, k data points are arbitrarily selected from the dataset as centroids. Euclidean distances of all the points from these centroids are calculated. Points closest to a particular centroid are then assigned to that particular cluster. The mean value of every cluster is calculated and the centroid is updated to that mean value. The data points are then re-assigned to their updated centroid values. This process is repeated until the shape of the cluster remains unchanged, that is, data points belonging to the same cluster do not get reassigned to a new cluster.

The key to achieve good clustering in our case, i.e. clusters which are representative of the sub-genres in the given collections of tunes, is to identify the features which comparison across tunes according to the euclidean distance is most representative of their belonging to a sub-genre.

Since several sets of features are being considered, we also test a slight modification of the K-Means clustering algorithm, weighted K-Means that is suitable for clustering items represented by multiple vectors. While in standard K-Means, the vectors would be concatenated (aggregated into one unique vector) to calculate an overall euclidian distance, in weighted K-Means, a weight is provided to represent the contribution of each of the vectors to the euclidian distance comparison of each item. This enables putting more or less importance to each set of features to guide the clustering mechanism.

The most effective clustering leads to minimum intra-cluster variability and maximum inter-cluster variability. We use Silhouette Analysis to evaluate the resulting clusters. In silhouette analysis, for every point i belonging to a cluster C, the mean distance  $a_i$  between i and all the points in C and the minimum distance  $b_i$  between i and any point outside of C are calculated. The silhouette coefficient S(i) of an item i is then given by:

$$S(i) = \frac{b_1 - a_i}{max(a_i, b_i)} \tag{1}$$

Silhouette coefficients are calculated for all the points present in the dataset and these values are averaged to get an overall result. Values of Silhouette coefficients can range from -1 to 1. Higher values are representative of better quality clusters.

## 6 Results

Various features were considered as input to the unsupervised learning model individually as well as in combination. The features giving the best results were finally consolidated in a single dataset. Two of the features, note duration and note pitch, however required to first be transformed into representations that were suitable for comparison through euclidian distance.

## 6.1 Representing pitch

The first set of features considered were the pitch values of the notes. For creating a vector of pitch values four different approaches were considered:

Approach 1: Vector of pitch values: In this approach, the pitch values were selected as is, i.e. the vector consists of pitch values in the way they appeared in

the MIDI files. For example, from a MIDI file including 10 notes ranging from A4 (69) to E5 (76), the vector representation might be:

[69, 69, 74, 74, 74, 73, 74, 76, 76, 76]

Approach 2: Vector of differences with first pitch value: In this approach, to give more of a notion of the progression of notes rather than of exact values, the first note (pitch value) was subtracted from all the notes. The vector, always starting at 0, is therefore a vector of pitch differences with the first note. Considering the pitch values of the example in Approach 1, the resulting vector would be:

[0, 0, 5, 5, 5, 4, 5, 7, 7, 7]

Approach 3: Vector of differences with the mean pitch value: Similarly to above, in this approach, the vector is created from the differences between each note's pitch value and the average pitch value. For the same example vector of pitch value, the result would therefore be:

$$[-4.5, -4.5, 0.5, 0.5, 0.5, -0.5, 0.5, 2.5, 2.5, 2.5]$$

Approach 4: Vector of differences with the previous note's pitch value: Finally, to represent a similar notion of progression which is less dependent on the overall tune, and more on the local changes in pitch in the tune, in this approach, the vector is made of the difference between the pitch value of the current note, with the pitch value of the previous note. On the same example, the result would be:

[0, 5, 0, 0, -1, 1, 2, 0, 0]

## 6.2 Representing duration and timing

Similarly to the pitch feature, two different approaches were considered for a representation of note duration.

Approach 1: Vector of note durations: In this approach, the duration of the notes in number of ticks as directly extracted from the MIDI file, is used as a vector. For example, for a tune including 10 notes, some being only 1 tick long, and some being up to 8 ticks long, the following vector might be used:

Approach 1: Binary vector of note hits: Since the duration vectors such as the one above might be difficult to meaningfully compare, we considered a different approach representing the notes' timing using a binary vector where each value represents a tick, and is equal to 1 if a note started playing on that tick, or 0 otherwise. Considering the smaller example of vector of durations:

[8, 8, 8, 4, 4]

the resulting binary vector would be:

[1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0]

## 6.3 K-Means clustering

We applied the K-Means algorithms on 400 tunes from the session, belonging to the Jigs, Reels, Polkas, and Barn dances genres with k = 4, using each of the features mentioned in the previous sections separately (cropping each vector to the length of the shortest tune in the collection). The average silhouette coefficient obtained are presented in Table 1.

Feature	K-Means silhouette coefficient
Pitch values	
A1: Pitch values	-0.002
A2: Difference with first	-0.002
A3: Difference with mean	0.03
A4: Difference with previous	-0.002
Duration and timing	
A1: Note duration	-0.07
A2: Binary note hits	0.20
Beats	0.159
Statistical Features	
Distribution of pitch values	0.0031
Distribution of velocity values	0.0036
MFCCs	-0.08

 Table 1. Silhouette coeficients of K-Means clustering using individual feature sets.

From Table 1, it can be observed that comparing pitch values by finding the difference with the mean of all notes, the timing of notes in a binary form, and beats give the best results in terms of average silhouette coefficients. Thus, these features were used in combination to test on the 32 selected tunes that overlap between ITMA and in the Session, with the results shown in Table 2. The combination here was achieved by representing each tune as a concatenated set of vectors, adding 0 padding to the shorter ones in order to have equal weight for each of the feature sets. As can be seen, results obtained from the combination of features are consistent, i.e. slightly better than the best feature individually. They also fall within the same range between the two datasets.

 Table 2. Results of combining the three selected feature sets on the overlapping sets of tunes from the Session and ITMA.

Dataset	Average silhouette coefficient
The Session Tunes	0.23
ITMA Tunes	0.22

### 6.4 Weighted K-Means clustering

As mentioned previously, considering the difference in their individual performances, better results might be obtained by giving more or less importance to each of the three selected feature sets. We therefore applied a Weighted K-Means algorithm where each of the feature sets is assigned a weight which corresponds to the contribution of the euclidian distance comparison of that feature set to the overall distance used in the comparison of two items.

**Table 3.** Results (average silhouette coefficient) of clustering with different weights associated with the pitch  $(w_p)$ , timing  $(w_t)$  and beats  $(w_b)$  feature sets.

Weights	Session dataset	ITMA dataset
$w_p = 0.33, w_t = 0.33, w_b = 0.34$	0.31	0.19
$w_p = 0.20, w_t = 0.20, w_b = 0.60$	0.22	0.03
$w_p = 0.45, w_t = 0.45, w_b = 0.10$	0.17	0.25
$w_p = 0.55, w_t = 0.25, w_b = 0.20$	0.31	0.19
$w_p = 0.10, w_t = 0.10, w_b = 0.80$	0.33	0.21

We systematically tested combinations of the weights  $w_p$ ,  $w_t$  and  $w_b$  for the pitch, timing and beats feature sets respectively. Table 3 presents silhouette scores in the two datasets for a sample of those combinations which demonstrate the range of results obtained. The best results in this table are also the best results overall. As can be seen, it is therefore possible to obtain better results with the weighted K-Means approach than with the base one. While the results are promising and show a reasonable ability of the clustering mechanism to distinguish groups of tunes that are expected to correspond to genres, the two different datasets perform differently, and achieve best results on different sets of weights. This is especially surprising as the two datasets contain the same set of tunes, from two different collections. Those tunes however are represented differently and have been transcribed in different conditions, showing how those aspects cannot be neglected when using automatic processes in ethnomusicology.

## 7 Conclusion

In this paper, we have explored the features and their representation that can support comparing traditional Irish tunes for the purpose of sub-genre identification. We applied clustering methods on tunes from two large collections of tunes, represented in MIDI files and ABC notations, to show that specific representations of the timing, beats and pitch envelope of a tune provide better results, especially when combined and weighted. Those promising results provide a better understanding of approaches that can be used to explore the delineation of genres and the relation between computable features and the perception music. They also provide a basis for music information retrieval applications that apply to traditional Irish music, and potentially beyond. Indeed, as shown in Figure 2, as a next step in this direction, we have built a prototype application enabling the user to retrieve tunes from the Session or ITMA based on their similarity to a given tune, using the representations established in this paper. Such an application has great potential for supporting music practitioners and researchers in the field.

## Sub-Genre Detection in Music Scores

Irish Traditional Music Archive <sup>●</sup> TheSession.org					
america	Submit				
Track-Name: "My love is in America" Genre:"Reel"					
▶ 0:00 / 0:56 ●	● :				
Top ten similar songs are:					
Track Name	Genre	Play Tune			
Bean dubh an ghleanna	Reel	► 0:00 / 1:38 ●			
The morning star	Reel	▶ 0:00 / 0:57 ● ● ▶ :			
The heather breeze	Reel	▶ 0:00 / 0:26 ● ▲ ►			

Fig. 2. Screenshot of prototype similarity-based tune search application.

# References

- 1. J Stephen Downie. Music information retrieval. Annual review of information science and technology, 37(1):295–340, 2003.
- John A Hartigan and Manchek A Wong. Algorithm as 136: A k-means clustering algorithm. Journal of the Royal Statistical Society. Series C (Applied Statistics), 28(1):100–108, 1979.
- 3. Kyuwon Kim, Wonjin Yun, and Rick Kim. Clustering music by genres using supervised and unsupervised algorithms. Technical report, Stanford University, 2015.
- J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3), 1998.
- 5. M. Lesaffre, L. Voogdt, M. Leman, B. Baets, H. Meyer, and J. Martens. How potential users of music search and retrieval systems describe the semantic quality of music. *Journal of the American Society for Information Science and Technology*, 59(5), 2008.
- T. Lidy, C. Silla, O. Cornelis, F. Gouyon, A. Rauber, C. Kaestner, and A. Koerich. On the suitability of state-of-the-art music information retrieval methods for analyzing, categorizing and accessing non-western and ethnic music collections. *Signal Processing*, 90(4), 2010.

- 12 G. Shingte and M. d'Aquin
- Cynthia Liem, Meinard Müller, Douglas Eck, George Tzanetakis, and Alan Hanjalic. The need for music information retrieval with user-centered and multimodal strategies. In MM'11 - Proceedings of the 2011 ACM Multimedia Conference and Co-Located Workshops - MIRUM 2011 Workshop, MIRUM'11, 2011.
- 8. Noris Mohd. Norowi, Shyamala Doraisamy, and Rahmita Wirza. Factors affecting automatic genre classification: An investigation incorporating non-western musical forms. In *Proc of ISMIR 2005, 6th International Conference on Music Information Retrieval*, 2005.
- S. Oramas and O Cornelis. Past, present, and future in ethnomusicology: The computational challenge. In *International Society for Music Information Retrieval*, 2012.
- Wei Peng, Tao Li, and Mitsunori Ogihara. Music clustering with constraints. In Proc. of ISMIR 2007, the 8th International Conference on Music Information Retrieval, ISMIR 2007, 2007.
- 11. L. Weissenberger. When "everything" is information: Irish traditional music and information retrieval. In *iConference*, 2014.