Classification of musical genre: a machine learning approach

Alfredo Serafini

CLASSIFICATION OF MUSICAL GENRE: A MACHINE LEARNING APPROACH

Roberto Basili, Alfredo Serafini, Armando Stellato
University of Rome Tor Vergata,
Department of Computer Science,
Systems and Production,
00133 Roma (Italy)
{*basili,serafini,stellato*}*@info.uniroma2.it*

ABSTRACT

In this paper, we investigate the impact of machine learning algorithms in the development of automatic music classification models aiming to capture genres distinctions. The study of genres as bodies of musical items aggregated according to subjective and local criteria requires corresponding inductive models of such a notion. This process can be thus modeled as an example-driven learning task. We investigated the impact of different musical features on the inductive accuracy by first creating a medium-sized collection of examples for widely recognized genres and then evaluating the performances of different learning algorithms. In this work, features are derived from the MIDI transcriptions of the song collection.

1. INTRODUCTION

Music genres are hard to be systematically described and no complete agreement exists in their definition and assessment. ”*Genres emerge as terms, nouns that define recurrences and similarities that members of a community make pertinent to identify musical events*” [11], [5].

The notion of community here play the role of a self-organizing complex system that enables and triggers the development and assessment of a *genre*. Under this perspective, the community plays the role of establishing an ontology of inner phenomena (properties and rules that make a genre) and external differences (habits that embody distinguishing behavior and trends).

In Information Retrieval the fact that relevance and relatedness are not local nor objective document properties but global notions that emerge from the entire document base is well known. Every quantitative model in IR rely on a large number of parameters (i.e. *term weights*) that in fact depend on the set of *all* indexed documents. It seems thus critical to abandon static ”grammatical” definitions and concentrate on representational aspects in forms of projections and cuts over the cultural hyperplane [1]. These aspects should not be postulated *a priori*, but acquired through experience, that is from living examples, of class membership.

For the above reasons, our analysis here concentrated on symbolic musical aspects so that as much information as possible about the dynamically changing genres (the target classes) could be obtained without noise (i.e. irrelevant properties implicit in the full audio content). Moreover, the analyzed features are kept as general as possible, in line with similar work in this area [13]: this would make the resulting model more psychologically plausible and computationally efficient.

Six different musical genres have been considered and a corpus of 300 midi songs – balanced amongst the target classes – has been built [1]. Supporting technologies ([3], [4]) have being employed to project relevant features out from the basic MIDI properties or from their XML counterpart ([12] [7] [14]). Machine Learning algorithms have been then applied as induction engines in order to analyze the characteristics of the related feature space. Although the study reported here is our first attempt to apply an inductive genre classification approach by exploiting MIDI information, our current work is also investigating audio properties over the same song collection.

2. SYMBOLIC REPRESENTATION OF MUSICAL INFORMATION FOR GENRE DETECTION

Previous work on automatic genre classification ([13]) suggests that surface musical features are effective properties in reproducing the speed of effective genre recognition typical of humans subjects. In a similar line we aim at determining a suitable set of features that preserve such accuracy over different and more fine-grain classes. Real genre classification require in fact more subtle distinctions and more insight is needed on the robustness of the inductive models with respect to this aspect.

[1] The corpus has been made freely downloadable at http:/ai-nlp.info.uniroma2.it/musicIR/MIDI_CORPUS_ISMIR04.zip

2.1. Coarse-Grain Features Definition

In this work aspects as melody, timbre and rhythm of a musical piece have been modeled by a small core of five coarse-grain feature classes. An evaluation of the effectiveness of very naif features as extracted from MIDI files is, in fact, needed to better assess the role MIDI could have in symbolic music retrieval. While melodic and rhythmic information are directly provided by MIDI files (e.g. note, metrics), ”voices” (i.e. patches) can be used as timbre properties.

Melodic Intervals All the basic melodic intervals within an octave are considered as a numeric feature: legal values indicate the relative frequency for each different melodic interval within the MIDI song.

Instruments The 128 patches of the General Standard Midi patch set surrogates the notion of instrument timbres.

Instrument Classes and Drumkits Each GSM patch is associated to exactly one of the common sixteen different instrument classes (i.e. Piano-like instruments, Strings, Synth Pads, Brass and so on). For drums, we considered the 8 different drumsets always associated with the midi channel 10. The different classes are here expressed as boolean features.

Meter/Time Changes Two numeric attributes represent respectively the mean *metronome time* and the number of different *meter/time* changes.

Note Extension Three features express the lowest, the highest and the global pitch extension of a piece. These features were introduced looking at the popular music octaves extension, which is typically tonally restricted (see also [11] about the Muzak phenomenon).

One of the aims of this research is to study the impact of simple features on genre classification. Although a wider set of properties can be easily derived, at this stage of the study, we mainly expect the machine learning algorithm to restrict the set of useful properties *against* the training data. These latter will be discussed in the next sections.

2.2. Corpus Construction

Our dataset includes about 300 midi files collected from the Web. The songs are clustered into six different musical genres, in order to have wide coverage of heterogeneous musical material and looking at music distribution and e-commerce definitions (e.g. `www.amazon.com`). To give a measure of the inherent complexity of the categorization task, we asked two annotators to annotate a large portion of the entire corpus. About 171 files have thus been independently assigned to one of the genres by each annotator. Then we computed a standard F-measure as a measure of the inter-annotator agreement (according to [9]) and we find a value of 0.85. For example, the results (in table 1) suggest a large disagreement for the *Pop* genre: this seems to confirm the common idea (see [11] [5]) that *Pop* music is a “mental melting pot” for songs that are not deeply rooted within a particular style, but better embraces the generic definition of ”common music appreciated by the mass”.

MusicalGenres	Annotations		Common Annotations	F-Measure
	1st	2nd		
Blues	51%	40%	40%	89%
Classical	17%	17%	17%	100%
Disco	31%	24%	24%	89%
Jazz	24%	28%	23%	89%
Pop	26%	29%	20%	73%
Rock	22%	33%	22%	83%

Table 1. F-measure between annotations amongst different musical genres

2.3. Machine Learning Algorithms

All the experiments have been run within the Waikato Environment for Knowledge Analysis (WEKA, ref. [6]). Various learning algorithms have been considered for our experiments, including decision-tree, Bayesian and rule-based classifiers:

The **Naive Bayes** classifier performs statistical analysis of the training data, produces maximum likelihood estimators and maximizes conditional probabilities on the observed feature values as decision criteria.

The **VFI** (Voting Feature Intervals) algorithm classifies by attribute-discretization: the algorithm first builds feature intervals for each class and attribute, then uses a voting strategy to assess its learning model. Entropy minimization is always used to create suitable intervals.

J48 is an implementation of the well-known Quinlan algorithm (as C4.5, [2]). This classifier builds a decision tree whose nodes represents discrimination rules acting on selective features. Classification reduces to top-down navigation, i.e. rule cascade: musical genres are triggered when leaves in the model tree are reached.

Strictly related to J48 it is the **PART** algorithm. It exploiting separate-and-conqueror strategies to select the best leaf at each iteration, thus building an optimized partial decision tree.

NNge (Nearest-neighbor-like algorithm using non-nested generalized exemplars), it’s a rule based classifier. It builds a sort of “hypergeometric” model, including if-then rules.

The last algorithm is RIPPER (**JRip**) a rule-based classifier that implements a propositional rule learner. The learning model is developed by iteration over a training subset, and by doing structure optimization (i.e. pruning) to minimize error rate. Details on the learning strategies and their implementation can be found in [6].

3. EXPERIMENTAL RESULTS

3.1. Experiments overview

Experimental evaluation has been carried out by partitioning the corpus in training and testing portions and using progressively smaller percentages of the training data (90%, 75%, 66%). Dynamic partitioning, with 5, 10, and 20 fold cross-validation has been also applied.

Two categorization models have been studied:

Single Multiclass Categorization: all instances are used and assignment to one of the six musical genres decided.

Multiple Binary Categorization: different categorizers (one for each target genre) are derived by independent training processes [2]. Learning applies on positive examples as training instances of the target class C and a balanced [3] set of negative instances, randomly selected from other classes.

A global overview of the performances for multiclass categorization obtained under different training/testing conditions is reported in figure 1.

3.2. Multiclass Categorization Overview

Figure 1 shows that the most promising classifier is the Bayesian one. On the contrary, tree- or rule-based algorithms seems to have a minor impact on our little scheme of comparison. The outperforming results of the Naive-Bayesian classifier (with respect to other types of algorithms) could be explained by the overall heterogeneity of features across the different examined classes. Rule or tree-based approaches, in fact, tend to cluster the truly discriminatory features to produce their classifiers and impose, in this way, a generalization over the features.

As confirmed in Table 2, recognition of $Classical$ music is the easiest sub-task, followed by $Jazz$ recognition. This latter is probably deeply characterized by the kind of adopted instruments sets as well as by its harmonic/melodic nature and syncopated rhythms.

A detailed study of the harmonic and melodic properties of musical pieces as well as the recognition of complex melodic, harmonic and rhythmic patterns on a larger scale would be very interesting over these two genres. For example, some of the errors depend on melodic intervals that are not important in terms of their frequency but according to their *contextual* characteristics, e.g. sets, patterns of occurrences of intervals as well as their joint distribution in a song. It must be noted that only melodic intervals are currently taken into account while harmonic properties are neglected. Vertical analysis would be certainly useful and it will be the target for future studies, as also suggested in [3] and [8].

3.3. Binary Categorization Overview

In figure 2 the performance of the 6 binary classifiers are reported comparatively with the performance of the multiclass classifier (column 1, *Multiclass*).

As expected, the binary classification outperforms the multiclass in terms of accuracy. The task of separating musical instances of a particular genre from all the others seems easier. However, current performances are good for genres which are different from the typical (and complex) structure of "Popular music" ([5]), e.g. $Classical$. $Jazz$ and $Rock$ series have a behavior closer to that of Pop.

[2] The results for binary models can be collected on a hierarchical meta-learner.

[3] i.e. a set of equivalent size, in order to balance the negative evidence within training and testing.

3.4. Analysis of Feature Classes

In table 2, the relative impact of each class of features on the classification accuracy are shown by using a Naive-Bayes Classifier as reference model.

Features	Precision	Recall	F-measure
Instruments (I)	72%	72%	71%
Instruments Classes (IC)	61%	64%	61%
M/K Changes (MKC)	41%	39%	34%
Melodic Intervals (MI)	36%	32%	25%
Notes Extension (NX)	26%	16%	16%

Table 2. Performance of Naive-Bayes Classifiers trained over different feature classes

As expected, the "*Instruments and Drumkits*" features is the most effective.

3.5. Performance Analysis

$Jazz$ and $Blues$ classifiers are often misleading each other: when $Jazz$ has low precision, the recall related to $Blues$ goes down. This reflects the fact that in a multiclass categorizer a class, though being easily recognizable by itself, is shaded by the similar characteristics of more prominent classes. Notice how $Blues$-$Jazz$ are also ambiguous for human annotators: this is probably inherent to the mutual ambiguity that characterizes these two genres. Following this observation, a comparative analysis of the differences between typical errors done by humans and machines could help in stressing which are the intrinsic and extrinsic properties of a musical piece and how they can help in recognizing its musical genre.

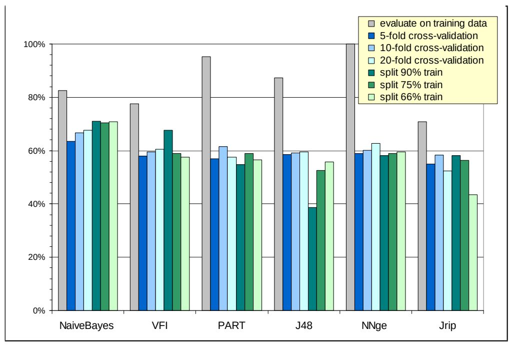

Figure 1. Multiclass Genre Classification: Performance evaluation of six different algorithms against different strategies of testing

In our experiments, we voluntarily limited our scope of investigation only to intrinsic properties of musical pieces, ignoring other (though important) informational resources like authorship, cultural context and release date. We reserve comparative studies on the above features for future research work.

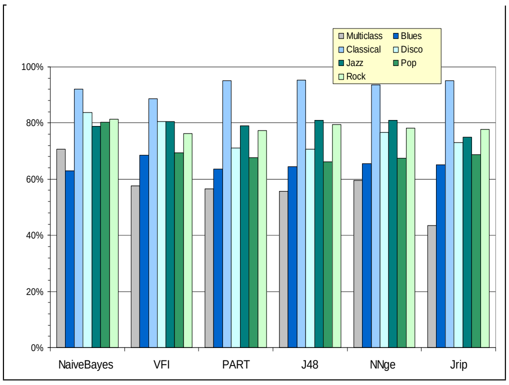

Figure 2. Binary Genre Classification: Comparizons between Algorithms Using 66% Training Set

4. CONCLUSION AND FUTURE WORK

The ambiguity inherent to every definition of Musical Genre, together with the high dynamics that undermines its persistency over time, characterizes the complexity of the automatic genre categorization task.

The idea that, neglecting absolute and general hypothesis and postulates about musical genres, these latter are to be explored, learned and recognized only through labeled examples, guided our investigation. Musical Genres can thus be redefined and tailored according to particular aspects of the domain of interest and to the degree of granularity they are supposed to bring in any given application.

Machine learning techniques have been applied to study the discriminatory power of different surface features, i.e. melodic, rhythmic and structural aspects of songs derived from their MIDI transcriptions. This task is necessary in view of multi-modal symbolic music analysis over heterogeneous representations (e.g. MIDI, MP3, xml descriptors such as musicxml).

Results are very encouraging. The performance of the automatic classifiers are comparable with those obtained in previous studies (over less granular categories, e.g. [13]). This suggests that simple musical features can provide (at a first level of approximation) effective information for genre categorization. The complexity of some subtasks (e.g. distinction between closer genres like *Jazz* and *Blues*) require more complex features, like vertical analysis.

This study represents an initial exploration in symbolic music feature analysis: other and more complex feature sets will be taken into account to build computational models better suited to recognize smaller differences between styles and genres. Our medium term target is also the realization of sensibly larger musical corpora, with different dimensions, class granularity and coverage. Large scale resources are in fact necessary to support more systematic experiments and assess comparative analysis. The collection adopted in this paper can be seen as a first resource for supporting the benchmarking of music categorization systems.

5. REFERENCES

[1] Ullmann, S. *Semantics: An Introduction to the Science of Meaning*. Oxford: Basil Blackwell, 1972.

[2] R. Quinlan, *C4.5 Programs for Machine Learning*, Morgan Kauffman, San Mateo, CA, 1993.

[3] Huron, D. *The Humdrum Toolkit: Software for Music Researchers*. Stanford, CA: Center for Computer Assisted Research in the Humanities, 1993.

[4] Clark, J. “XSL Transformations (XSLT) Version 1.0”, *http://www.w3.org/TR/xslt*

[5] Fabbri, F. “Browsing Music Spaces: Categories And The Musical Mind”, *http://www.mediamusicstudies.net/tagg/others/ /ffabbri9907.html*

[6] Witten, I.H. Frank, E. Kaufmannm M. *Data Mining: Practical machine learning tools with Java implementations*. www.cs.waikato.ac.nz/˜ ML/weka/book.html, San Francisco, 2000.

[7] Cover, R. “The XML Cover Pages: XML and Music”, http://xml.coverpages.org/xmlMusic.html

[8] Manaris, B. Sessions, V. Wilkinson J. “Searching for Beauty in Music-Applications of Zipf’s Law in MIDI-Encoded Music”, *ISMIR 2001*

[9] Boisen, S. Crystal, M. Schwartz, R. Stone, R. and Wischedel, R. “Annotating Resources for Information Extraction, *LREC 2002 pp. 1211-1214*

[10] Aucouturier, J.J. Pachet, F. “Representing musical genre: A state of the art.”, *Journal of New Music Research, 32(1):83-93*

[11] Fabbri, F. *Il Suono in cui Viviamo*. Arcana, Italy, 2002.

[12] Hinojosa Chapel, R. “XML-Based Music Languages A Brief Introduction”, *Pompeu Fabra University, Barcelona, Spain, 2002*

[13] Tzanetakis, G. Cook, P. “Musical genre classification of audio signals.”, *IEEE Transactions on Speech and Audio Processing, 10(5)*

[14] *MusicXML Definition*, Version 1.0: January 13, URL: http://www.musicxml.org/xml.html. 2004.