CLASSIFICATION INTO MUSICAL GENRES USING A RHYTHMIC KERNEL

Gustavo Frederico
Université d'Ottawa
École d'Ingénierie et de
Technologie de l'Information
800 Avenue King Edward
P.O. Box 450, Succ. A
Ottawa, Ontario
K1N 6N5 Canada
gfred006@uottawa.ca

ABSTRACT

Beginning with the question on how to determine the genre of a music piece, we elaborate on the representation of rhythm for the classification into genres. The aim of such classification differs in principle from that of traditional Music Information Retrieval algorithms. First, we formalise the rhythmic representation of music fragments. This formalism is then used to construct a similarity function called kernel. To allow the discrete comparison of rhythmic fragments, a pre-processing step in the algorithm computes a common quantization unit among the input data. A simple injective mapping into $\mathbb{R}^N$ allows the kernel to employ the Euclidean dot product. A small database of jazz, classical and rock fragments is used in an implementation of a Support Vector Machine. The issues that arise with different time signatures are analysed. Finally, we share some early results of the experiments comparing the three genres, showing that rhythm conveys good information for classification, within the conditions of the experiment.

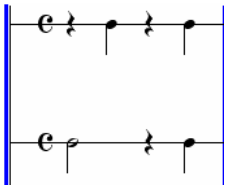
1. INTRODUCTION

The present work started when considering the question on how to determine the music genre of a music piece. Despite current inconsistencies with genre taxonomies in the music industry [8], there is informally sufficient agreement on the taxonomy of western genres that allow us to classify music pieces into broad and distinctive categories such as rock, blues, jazz and classical. Cognitively, it is relatively easy even for the untrained ear to make such distinction. Publication in the field of Music Information Retrieval (MIR) has been produced that tries to identify and retrieve music pieces or fragments [4, 11]. Meudic proposes similarity measurements for rhythm, pitches and contours [5, 6]. In many instances, the problem consists of finding the best ways to match a given (possibly "inaccurate") execution performed by a human against a collection of music pieces. We, in turn, shall concentrate on another problem, that of classifying existing musical pieces according to their genres. For instance, two distinct musical pieces might have very different melodies and have been composed by different authors. Despite these facts, they can still belong to the same genre. We are more interested in investigating the musical properties underlying the pieces that allow for the distinction and the formalism of the musical representation. We begin by limiting ourselves to the rhythmic content of musical pieces. This paper is organized as follows: section 2 describes the musical representation of rhythmic fragments. The mathematical representation is then used in the theory of Kernels in section 3. Section 4 describes the implementation of the Support Vector Machine and the early results of the experiment. We close laying out possible paths for future research in section 5 and reasoning upon the results in conclusion. Our hope is that the techniques employed while trying to answer this question shall evolve and collaborate along with other existing formalisms at some point in the future as tools of musical analysis.

2. RHYTHMIC REPRESENTATION

A polyphonic rhythmic fragment is defined as parallel sequences of note durations over time. We construct our set X of possible polyphonic rhythmic fragments formally as follows: consider the finite set $H \subseteq B^n$ of monophonic rhythmic fragments, where B = {sound, rest}. B is isomorphic to the Boolean set which, indeed, is the internal representation used in our experiment. H is a monoid with the binary operator being the note concatenation. $X \subseteq H^m$ is a monoid with the binary operator defined as voice concatenation.

For example, the polyphonic rhythmic fragment



is represented as the element {(rest, sound, rest, sound) ,(sound, sound, rest, sound)} $\in X$, taking the quarter note as the quantization unit. Differently than Vuza's rhythmic representation of periodic subsets of $\mathbb{Q}$ [10], ours is not concerned with the intrinsic periodicity of patterns within a given music piece.

The music pieces represented as elements of X serve as input to the Support Vector Machine that performs the learning and classification tasks.

We introduce one pre-processing step that adjusts the internal representation of the input due to possible differences in the quantization of elements of X. A quantization unit is a number of the format q^{-1}, $q \in \mathbb{N}^{+}$. Figure 1 shows two rhythmic fragments with two, *a)* and *b)*, with quantization units 4^{-1} and 6^{-1}, respectively. In this case, the common quantization unit is 12^{-1}. Generically we have

$$q = \frac{1}{lcm(\frac{1}{q_i})} \tag{1}$$

for the overall quantization q for all inputs. This pre-processing step will enable a fair comparison between rhythmic fragments in the algorithm. Differences in time signature will have certain implications in the comparisons, as discussed in section 4.

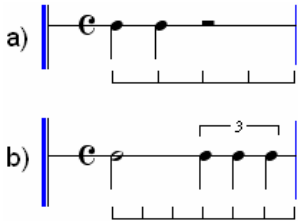


Figure 1: Two rhythmic fragments with different quantization

3. KERNELS

Kernels are functions that measure a degree of similarity between two objects. Kernels have been successfully used in pattern recognition problems, including identification of performers by their playing style [2] and text classification [1]. One of the main kernel algorithms is Support Vector Machines (SVMs). Extensive information about kernels and SVMs can be found in [1] and [9].

The learning process in SVMs itself takes place in two steps: the learning and the testing phases. In the learning phase, SVMs take pairs of the form $(x_i, y_i) \in X \times \{+1,-1\}$ where in our case $y_i \in \{+1,-1\}$ represents one of two different target genres for input sample x_i.

The algorithm then estimates a decision function $h: X \to \{+1,-1\}$ that generalizes for unseen data inputs while trying to minimize the risk of misclassification. The decision function h makes use of the similarity function $K: X \times X \to \mathbb{R}$, the kernel function. The kernel function measures the similarity between two input data. The decision function is used during the testing phase and has the form

$$\begin{aligned} h(x) &= sign(\sum_{j=1}^{\ell} \alpha_j y_j K(x_j, x) + b) \\ &= sign(\sum_{j=1}^{\ell} \alpha_j y_j \langle \mathsf{f}(x_j), \mathsf{f}(x) \rangle + b) \end{aligned} \tag{2}$$

where l is the size of the training sample set, f is a feature mapping described below and α_j is the embedding strength of input x_i. Musical instances that are more difficult to classify during the learning phase yield higher values in α by the algorithm. α and b are calculated during training. By convention, $sign(0)=+1$. In this domain, the objective in the design of the kernel function is to measure quantitatively genre similarity.

In certain problems, a mapping from the input space into another space, called the feature space, is explicitly defined. This occurs often under two circumstances: when the data is not linearly separable in the original input space or when the input space X does not allow a scalar multiplication $\mathbb{F} \times X \to X$, where $\mathbb{F}$ is one the possible kernel fields $\mathbb{R}$ or $\mathbb{C}$. The feature space is possibly high dimensional or infinite.

We define the feature mapping $\mathsf{f}: X \to \mathbb{R}^N$ as

$$\mathsf{f}(x) = (u_i) \tag{3}$$

where each component u_i of the vector is 1 if there is any sound at time slot i and 0 otherwise. Therefore, the image of f forms a spanning set of $\mathbb{R}^N$ where each component of each vector is either 0 or 1. The vertical projection introduced by f collapses notes from all lines into one voice. The result is a non-polyphonic representation in $\mathbb{R}^N$ that carries some polyphonic aspect of the whole piece but that lacks the completeness of the overall polyphonic information embedded in the original representation.

The kernel function is

$$\begin{aligned} K(x, x') &= \langle \mathsf{f}(x), \mathsf{f}(x') \rangle \\ &= \sum_i u_i u_i' \end{aligned} \tag{4}$$

that is the dot product in $\mathbb{R}^N$. The kernel is in fact computing the rhythmic similarity between two fragments by adding 1 if and only if there is a note in common between the fragments. The comparison is performed by the multiplication.

The feature mapping f is injective but not surjective. In fact, it is introduced here to comply with

the requirement of existence of a multiplication operation by a real scalar in a vector space over the real field. This multiplication operation is explicit in the inner product's linear function property

$$\langle au + bv, z \rangle = a\langle x, z \rangle + b\langle v, z \rangle \quad (5)$$

for all a,b $\in \mathbb{R}$, u,v $\in$ some feature space F, a Hilbert space. If X were an abelian group with a scalar multiplication $\mathbb{R} \times X \to X$ defined, and conformed to the requirements for becoming such Hilbert space, it would be possible to establish an isomorphism between X and $\mathbb{R}^N$. However, the purely rhythmic information contained in X excludes a meaningful semantic interpretation of what would be real numbers representing rhythmic events in the original input domain. The feature mapping is then explicitly declared, despite its injective property. Given the image of the feature mapping, the image of the inner product function is the non-negative natural numbers.

4. THE EXPERIMENT AND P RELIMINARY RESULTS

A kernel function defines a similarity function that can be used in different learning algorithms. Our implementation of the kernel K was linked to the SVMLight software [3], utilizing its user-defined kernel option. We categorized the pieces into three genres: jazz, rock and classical, performing the training and classification in pairs. This approach to multi-class classification, called pairwise classification, trains and executes one classifier for each pair of classes. For small number of genres pairwise classification is still practical. However at some point other approaches are necessary.

In the experiment, the musical pieces were arbitrarily selected, with some restriction with regards to the time signature. They are grouped by genre in table 1.

The theory allows for the kernel to compare fragments with different time signatures. However, the rhythmic fragments were deliberately restricted to those with time signatures 2/2, 2/4 and 4/4, with the intention of avoiding a vertical 'misalignment' of beats and a more complex interpretation of the results. For example, consider the two fragments in Figure 2.

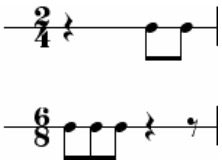


Figure 2: Two rhythmic fragments with different time signatures

They are modelled as {(rest, rest, sound, sound)} and {(sound, sound, sound, rest, rest, rest)}, respectively, taking the eight note as quantization unit. The current model would have counted one sound in common in the dot product, that in the 3rd slot. Another interpretation could use a different quantization formula aligning the bars, which would yield no sound in common in the dot product. Thus, the question of how to properly compare rhythmic fragments with different time signatures remains open.

Music	Composer	Measures
Classical		
Sonata No. 8 C minor, Op. 13 (Pathétique), Movement 2	Beethoven	23
The Well-Tempered Clavier Part I, Prelude and Fugue 5 in D major BWV 850	Bach	27
The Seasons, July. Song of the Reaper	Tchaikovsky	36
Symphony no. 25 in g minor, K 183, Movement 1	Mozart	34
Concerto No.1 in A Minor BWV1041	Bach	85
Partita for unaccompanied flute, Allemande BWV 1013	Bach	65
Rock		
Start Me Up	Rolling Stones	44
Eye Of The Beholder	Metallica	53
Stairway to Heaven	Led Zeppelin	65
Let it Be	Beatles	62
Just Got Paid	ZZ Top	53
Jazz		
On Green Dolphin Street	Ned Washington and Bronislaw Kaper	58
Lazy Bird	John Coltrane	52
Serial Number	Mark Kramer	58
One For Helen	Bill Evans	56
Speak Low	Kurt Weill	50

Table 1. Sample music by genre, composer and measurers

The classification accuracy represents how many music fragments were correctly classified over the total number of fragments pairwise. The accuracy of the SVM is depicted in table 2.

Albeit the relatively small database, one can see that the rhythmic kernel conveys good information for classifying musical genres. When interpreting the

numbers, one should keep in mind the limited number of samples, the treatment given to polyphony and the fact that only the rhythmic feature was used.

	Rock	Classical
Jazz	0.8	0.818
Rock		0.909

Table 2. Accuracy of the experiment

5. AREAS OF THEORETICAL INVESTIGATION

The bilinear form in K's inner product has codomain $\mathbb{R}$, despite its inherent discrete nature. This motivates a search for a discrete definition of a kernel. Ongoing research is trying to redefine the rhythmic representation so that an inner product applies directly to the input set and the feature mapping does not have to be explicit. The idea consists of finding a symmetric bilinear form whose domain is a $\mathbb{Z}$-module. Modules are generalizations of vector spaces. Instead of using the traditional definition of inner product over the field $\mathbb{R}$, this approach will try to find an inner product module over $\mathbb{Z}$. A free $\mathbb{Z}$-module is an inner product space [7].

Other areas of investigation would include the study of the influence of other features such as harmonic and intervallic structures in trying to classify music pieces according to their genre.

6. CONCLUSION

The rhythmic representation introduced here allows us to construct a kernel function that computes a similarity measurement that can be used in the categorization of musical genres. Although a relatively small database for testing was used, this formalism along with the categorization capabilities of Support Vector Machines shows that good classification accuracy can be achieved from the rhythmic structure of fragments of music pieces. Future research would allow a more explicit use of the polyphonic structure of pieces. A proper assessment on the generalization capability of the model is still necessary. The generalization allows one to bound the expected error rate in classification for a large number of rhythmic fragments given a certain number of fragments as training inputs.

7. REFERENCES

[1] Cristianini N. and Shawe-Taylor J., *An Introduction to Support Vector Machines*, Cambridge University Press, 2000.

[2] Hardoon, D. R., Saunders, C., Shawe-Taylor, J. and Widmer, G. "Using String Kernels to Identify Performers from their Playing Style", *Proceedings of European Conference on Machine Learning (ECML)*, Pisa, Italy, 2004.

[3] Joachims, T., SVM-Light, http://svmlight.joachims.org/.

[4] Lemström, K., Wiggins, G. A., Meredith, D., A "Three-Layer Approach for Music Retrieval in Large Databases." *ISMIR 2001*, Bloomington, Indiana, 2001.

[5] Meudic B., "Musical Pattern Extraction: From Repetition to Musical Structure", *Computer Music Modeling and Retrieval*, Montpellier, 2003.

[6] Meudic B., "Musical Similarity in a Polyphonic Context: A Model Outside Time", *Proceedings of the XIV Colloquium on Musical Informatics (XIV CIM 2003)*, Firenze, Italy, May 8-9-10, 2003.

[7] Milnor, J. and Husemoller, D., *Symmetric Bilinear Forms*, Ergebnisse der Mathematik und ihrer Grenzgebiete, Springer-Verlag, 1973.

[8] Pachet, F.; Cazaly, D., "A Taxonomy of Musical Genres." *Content-Based Multimedia Information Access Conference (RIAO)*, Paris, April 2000.

[9] Schölkopf, B. and Smola, A. J., *Learning with Kernels – Support Vector Machines, Regularization, Optimization and Beyond*, MIT Press, 2002.

[10] Vuza, D. T., "Sur le Rythme Périodique". *In M. Boroda (ed.) Quantitative Linguistics.* Musikometrika, 37, 1988.

[11] Welsh M. et al., *Querying Large Collections of Music for Similarity*, UC Berkeley Technical Report UCB/CSD-00-1096, November, 1999.