Classifier ensembles for genre recognition *

Pedro J. Ponce de León, José M. Iñesta, Carlos Pérez-Sancho
Universidad de Alicante.
Departamento de Lenguajes y Sistemas Informáticos
P.O. box 99, E-03080 Alicante, Spain
{*pierre,inesta,cperez*}*@dlsi.ua.es*
`http://grfia.dlsi.ua.es`

Abstract

Previous work done in genre recognition and characterization from symbolic sources (monophonic melodies extracted from MIDI files) have pointed our research to the use of classifier ensembles to better accomplish the task. This work presents current research in the use of voting ensembles of classifiers trained on statistical description models of melodies, in order to improve both the accuracy and robustness of single classifier systems in the genre recognition task. Different voting schemes are discussed and compared, and results for a corpus of Jazz and Classical music pieces are presented and assesed.

Keywords: Statistical pattern recognition, Classifier ensembles, Music information retrieval, Musical genre recognition

1 Introduction

Some recent works explore the capabilities of machine learning or pattern recognition methods to recognise music genre, either using audio [1, 2, 3], or symbolic [3, 4, 5] sources, or even metadata [6]. After a period of time doing research on the use of statistical models and classification paradigms for music genre (or style) characterization from symbolic data [7, 8], we reached a point where the combination of the different learning systems we developed showed up as the logical next step in our research. The many ways of building classifier *ensembles* (i.e., combining different classifiers) to improve both the accuracy and robustness of single classifiers is a hot topic in the areas of machine learning or pattern recognition. Works on this subject point out the importance of the concept of *diversity* in classifier ensembles, with respect to both classifier outputs and structure [9, 10, 11].

Our current research on combination of several previously developed classification systems for genre recognition in the symbolic domain is presented in this paper.

*This work was supported by the projects Spanish CICyT TIC2003–08496–C04, partially supported by EU ERDF, and Generalitat Valenciana GV043–541.

MIDI files have been used as the primary source of music data so, first, the music corpus of such files used is described. Second, the statistical description models utilized to describe music content are presented. Next, the classification techniques based on them are described, along with the different ensemble schemes for combining classifier decisions. Following this, the results for the ensembles are presented and compared with individual classifier results for genre recognition. Finally, the conclusions drawn from the results are discussed, pointing the research to further work lines.

2 Music data

The music corpus used is a set of MIDI files from *Jazz* and *Classical* music with a monophonic melody track, collected from different sources. No preprocessing of these files was done before entering the system, except for manually checking the presence and correctness of key, tempo, and meter meta-events, as well as the labeling of the melody track.

The corpus is made up of 110 files. 45 files are classical music files and 65 are jazz files, with a total length around 10,000 bars (more than six hours of music). The classification systems presented here work only on the information contained in the melody track. The rest of the MIDI file content is ignored because one of the general aims of this work is to analyze how much of the genre information is contained in the melody alone.

Two different ways of describing the content of the melody track have been used. The first one is based on melodic, harmonic, and rhythmic statistical descriptors and the second one describes melodic content in terms of strings of symbols corresponding to melody subsequences. Both description methods are briefly described in the following sections.

3 Statistical description models

3.1 Shallow structure descriptors

The first group of description models that have been used are based on descriptive statistics that summarise the content of a melody in terms of pitches, note durations, silences, harmonicity, rhythm, etc. This kind of statistical description of musical content is sometimes referred to as *shallow structure description* [12].

In these models, each melody is described by a vector of statistical descriptors, labeled with the genre of the melody. A set of 28 descriptors has been defined, based on several categories of features that assess melodic, harmonic, and rhythmic properties of a melody. These descriptors are summarized in Table 3.1. The first column indicates the musical property analysed and the other columns indicate the kind of statistics describing the property. A blank entry in the table means that a particular statistic has not been computed.

Four different description models have been defined. The model containing all the descriptors is called the F (full) model. From this one, three reduced models have been derived. This has been achieved using a per-feature separability test described in [13] to rank the features. Subsets of features are incrementally built by choosing the best ranked features. These models are called here A, B, and C for simplicity. Model A includes the six best ranked features, model B adds four features to model A, and model C adds two features to model B, so that $A \subset B \subset C \subset F$. Each entry in Table 3.1 indicates the smallest feature subset where the particular statistical descriptor has been included.

Category	Counter	Range	Avg. (relative)	Dev.	Normality
Notes	A				
Significant silences	B				
Non significant silences	F				
Pitches		A	A	A	F
Note durations		F	F	C	F
Silence durations		F	F	F	F
Inter-onset intervals		F	F	B	F
Pitch intervals		A	F	B	B
Non-diatonic notes	F		F	C	F
Syncopations	A				

Table 1: Shallow structure descriptors

For the descriptor computations, the melodies are quantized to a resolution of $Q = 48$ ticks per bar. Durations are measured in ticks. For pitch and interval categories, the range descriptors are computed as the maximum minus the minimum value in the melody, and the average-relative descriptors are computed as the average value minus the minimum value. For durations (note and silence durations, and inter-onset intervals) the range descriptors are computed as the ratio between the maximum and the minimum values, and the average-relative descriptors are computed as the ratio between the average and the minimum value. Finally, normality descriptors are computed using the D'Agostino statistic [14] for assessing the normality of the distribution of each property.

3.2 n-word based descriptors

The n-word based models make use of text categorization methods to describe melodic content. The technique encodes note sequences as character strings, therefore converting a melody in a text to be categorized. Such a sequence of n consecutive notes is called an n-word. All possible n-words in a melody are extracted, except those containing a silence lasting four or more beats. The encoding for n-words used in this work has been derived from the method proposed in [15]. This method generates n-words by encoding pitch interval and duration information. For each n-note sequence, all pitch intervals and duration ratios (inter-onset interval ratio) are calculated using Eqs. (1) and (2) respectively:

$$I_i = Pitch_{i+1} - Pitch_i \qquad (i = 1, \dots, n-1) \tag{1}$$

$$R_i = \frac{Onset_{i+2} - Onset_{i+1}}{Onset_{i+1} - Onset_i} \qquad (i = 1, \dots, n-2) \tag{2}$$

and each n-word is defined as a string of symbols:

$$[\ I_1 \ \ R_1 \ \ \dots \ \ I_{n-2} \ \ R_{n-2} \ \ I_{n-1} \ \ R_{n-1}\ ] \tag{3}$$

where the pitch intervals and duration ratios have been mapped into alphanumeric characters (see [8, 15] for details).

This method represents a musical piece as a vector $\mathbf{x}_i = \left(x_{i1}, x_{i2}, \dots, x_{i|\mathcal{V}|}\right)$, where each component represents the presence of the word w_t in the melody, being $|\mathcal{V}|$ the size of the vocabulary, that is, the total number of different n-words extracted from the corpus.

A common practice in text classification is to reduce the dimensionality of those vectors (usually very high) by selecting the words that contribute most to discriminate the class of a document (a melody here). The *average mutual information* measure (AMI) [16] has been used in this work to rank the words. This measure gives a high value to those words that appear often in melodies of one genre and are seldom found in melodies of the other genres. The n-words are sorted using this value, so only information about the first $|\mathcal{V}|$ words are provided to the classifier.

4 Classification techniques

4.1 Classifiers for shallow statistical features

Two different classification paradigms have been used with the four description models presented in section 3.1: the k-nearest-neighbour classifier, and the bayesian classifier assuming non-diagonal covariance matrices [17]. For the first one, given a sample $\mathbf{x}_i$, the distances to the prototypes in the training set are computed, and the class labels of the closest k are taken into account to take the decision by a majority. A value $k = 7$ has been establish for this classifier after some trials.

In the bayesian classifier the classification is performed following the well-known *Bayes' classification rule*. In a context where there is a set of classes $c_j \in \mathcal{C} = \{c_1, c_2, \ldots, c_{|\mathcal{C}|}\}$, a sample $\mathbf{x}_i$ is assigned to class c_j with maximum a posteriori probability, in order to minimize the probability of error:

$$P(c_j|x_i) = \frac{P(c_j)P(x_i|c_j)}{P(x_i)} \quad . \tag{4}$$

Using these two different classification techniques, eight different classifiers have been defined using the four shallow structure description models presented in section 3.1. Each classifier has been trained separately on the musical corpus and its accuracy estimated through leave-one-out cross-validation.

4.2 Naive Bayes classifier for n-words

For n-word based melody categorization, the naive Bayes classifier, as described in [18], has been used. Here, the classifier is based on the same Eq. 4, but applying the *naive Bayes assumption*, i.e. it is assumed that all words in a melody sample are independent of each other, and also independent of the order they were generated. This assumption is clearly false in our problem and also in the case of text classification, but naive Bayes can obtain near optimal classification errors in spite of that [19].

In this work, classes are musical genres, and the class-conditional probability of a melody $P(\mathbf{x_i}|\mathbf{c_j})$ is given by the probability distribution of note sequences (n-words) in genre c_j, which can be learned from a labeled training set, $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$. Two different distribution models have been used for the class-conditional probability: a Multivariate Bernoulli (MB) model, where the components of a sample vector are $x_{it} \in \{0, 1\}$ and a Multinomial (MN) model, where components are $x_{it} \in \{0, 1, ..., |\mathbf{x}_i|\}$, being $|\mathbf{x}_i|$ the number of n-words extracted from melody $\mathbf{x_i}$. Both MB and MN distributions have proven to achieve quite good results in text classification [18] and are briefly described below.

In the MB model, each class follows a multivariate Bernoulli distribution where the parameters to be learned from the training set are the class-conditional probability of each word in the vocabulary.

The MN model takes into account word frequencies in each melody, rather than just the occurrence or non-occurence of words, as in the MB model. In consequence, each component x_{it} is the number of occurrences of word w_t in the melody. In this model, the probability that a melody has been generated from a genre c_j is a multivariate multinomial distribution, where the melody length is assumed to be class-independent [18].

4.3 Classifier ensembles

After analysing the performance of the different classifiers studied, we have found a diversity of errors among the decisions taken by the different classifiers. This diversity has been suggested by some authors [10, 20] as an argument for using classifier ensembles with good results. These ensembles could be regarded as committees of 'experts' [21] in which the decisions of individual classifiers are considered as opinions supported by a measure of confidence usually related to the accuracy of each classifier. The final classification decision is taken either by majority vote or by a weighing system.

4.3.1 Voting schemes.

Designing a suitable method of decision combination is a key point for the ensemble's performance. In this paper, different possibilities that are presented below have been explored and compared. In the discussion that follows, N stands for the number of samples contained in the training set $\mathcal{X} = \{\mathbf{x}\}_{i=1}^{N}$, M is the number of classes in a set $\mathcal{C} = \{c_j\}_{j=1}^{M}$, and K classifiers, C_k, are utilized.

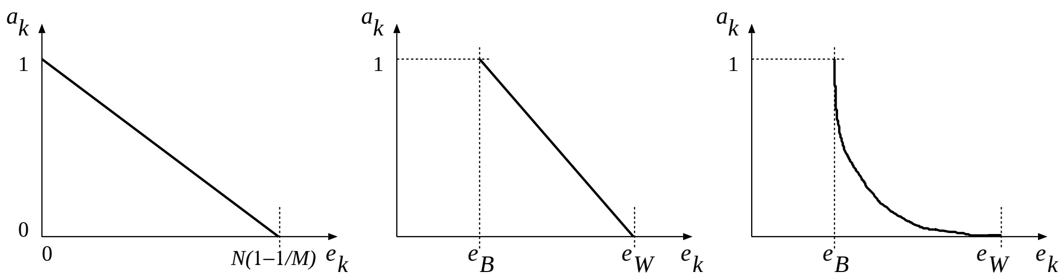

Figure 1: Different models for giving the authority (a_k) to each classifier in the ensemble as a function of the number of errors (e_k) made on the training set.

1. Majority vote. This is the simplest method. It just counts the number of decisions for each class and assigns the sample $\mathbf{x}_i$ to the class c_j that obtained the highest number of votes. The snag here is that all the classifiers have the same 'authority' regardless of their respective abilities to classify properly. In terms of weights it can be considered that $w_k = 1/K \ \forall k$.

2. Simple weighted majority. The decision of each classifier, C_k, is weighed according to its estimated accuracy (the ratio of successful classifications, α_k) on the training set [22]. This way, the authority for C_k is just $a_k = \alpha_k$. Then, its weight w_k is:

$$w_k = \frac{a_k}{\sum_l a_l} \quad . \tag{5}$$

Also for the rest of weighting schemes presented here, the weights are the normalized values for a_k, as shown in this equation.

The weak point of this scheme is that an accuracy of 0.5 in a two-class problem still has a fair weight although the classifier is actually unable to predict anything useful. This scheme has been used in other works [23] where the number of classes is rather high. In those conditions this drawback may not be evident.

3. Re-scaled weighted majority. The idea is to assign a cero weight to classifiers that only give N/M or less correct decisions on the training set, and scale the weight values proportionally, assigning $a_k = 1$ to the perfect classifier. As a consequence, classifiers with an estimated accuracy $\alpha_k \leq 1/M$ are actually removed from the ensemble. The values for the authority are computed according to the line displayed in figure 1-left. Thus, if e_k is the number of errors made by C_k, then

$$a_k = \max\{0, 1 - \frac{M \cdot e_k}{N \cdot (M-1)}\} \quad .$$

4. Best-worst weighted majority. In this ensemble, the best and the worst classifiers in the ensemble are identified using their estimated accuracy. A maximum authority, $a_k = 1$, is assigned to the former and a null one, $a_k = 0$, to the latter, being equivalent to remove this classifier from the ensemble. The rest of classifiers are rated linearly between these extremes (see figure 1-center). The values for a_k are calculated as follows:

$$a_k = 1 - \frac{e_k - e_B}{e_W - e_B} \quad ,$$

where $e_B = \min_k\{e_k\}$ and $e_W = \max_k\{e_k\}$.

5. Quadratic best-worst weighted majority. In order to give more authority to the opinions given by the most accurate classifiers, the values obtained by the former approach are squared (see figure 1-right). This way,

$$a_k = (\frac{e_W - e_k}{e_W - e_B})^2 \quad .$$

4.3.2 Classification.

Once the weights for each classifier have been computed, the class receiving the highest score in the votation is the final class prediction. If $\hat{c}_k(\mathbf{x}_i)$ is the prediction of C_k for the sample $\mathbf{x}_i$, then the prediction of the ensemble can be computed as

$$\hat{c}(\mathbf{x}_i) = \arg\max_{c_j \in \mathcal{C}} \sum_k w_k \delta(\hat{c}_k(\mathbf{x}_i), c_j) \quad , \tag{6}$$

being $\delta(a, b) = 1$ if $a = b$ and 0 otherwise.

Since the weights represent the normalized authority of each classifier, it follows that $\sum_{k=1}^{M} w_k = 1$. This makes possible to interpret the sum in Eq. 6 as $P(c_j|\mathbf{x}_i)$, the probability that $\mathbf{x}_i$ is classified into c_j, and $\hat{c}(\mathbf{x}_i)$ as the class for which this probability is maximum.

5 Results

The classifiers described in sections 4.1 and 4.2 have been utilized in order to build the ensembles, combining the different description models and classification paradigms: four k-nearest neighbors, using $k = 7$, with the different feature combinations (A, B, C, and F models), four Bayesian classifiers with the same feature combinations, and two naive Bayes using Bernoulli and Multinomial probability distributions. For the latter, a vocabulary size of 100 and 170 2-words have been used respectively, according to their AMI values. This makes a total of ten classifiers for building ensembles. Table 5 presents the estimated accuracy of the individual classifiers, α_k, obtained using a leave-one-out validation method on the training set.

Five different ensembles have been constructed using the five different votation methods described above (represented here as V1, V2, V3, V4, and V5). The decisions of the ensembles are summarised in Table 5 (*# errors all* column), and graphically depicted in Fig. 2 against the best individual classifier score. Note that the ensemble's performance using the quadratic best-worst strategy improves the behaviour of the best of the individual classifiers: just two errors against the three

Classification paradigm	Statistical model	Feature selection	# errors	α_k
7-nearest neighbours	Shallow	A	7	0.936
	Shallow	B	12	0.891
	Shallow	C	12	0.891
	Shallow	F	3	0.973
Bayes	Shallow	A	10	0.909
	Shallow	B	9	0.918
	Shallow	C	10	0.909
	Shallow	F	22	0.746
Naive Bayes	Bernoulli	$\|\mathcal{V}\| = 100$	8	0.923
	Multinomial	$\|\mathcal{V}\| = 170$	16	0.855

Table 2: Working parameters and accuracy of the different classifiers selected.

Voting method	# errors all	%	# errors all-but-best	%
V1	6	94.5	5	95.5
V2	9	91.8	9	91.8
V3	5	95.5	8	89.1
V4	3	97.3	4	96.4
V5	2	98.2	4	96.4

Table 3: Ensemble's performance.

errors made by 7-nearest neighbour classifier based on the whole set of shallow descriptors. Also it is interesting to see that majority voting and simple or re-scaled weighted majority perform clearly worse than the best-worst scale-based schemes.

The question arises of how sensitive is this success to the construction of the ensemble. In addition, is it worth to build an ensemble for avoiding just one error? The answer for both questions could be approached removing from the ensemble the best of the classifiers and analysing how much the performance is degraded. Thus, the 7-nearest neighbour classifier trained with the F model was dropped from the ensemble, and the new results were those also shown in Table 5 (*# errors all-but-best* column).

Note how, although the results are not as good as earlier, some ensembles mantain a high standard of precision, with just 4 errors. This clearly improves the performance of the current best classifier (7 errors), so the ensemble seems quite robust and performs well, specially with the best-worst strategies introduced here.

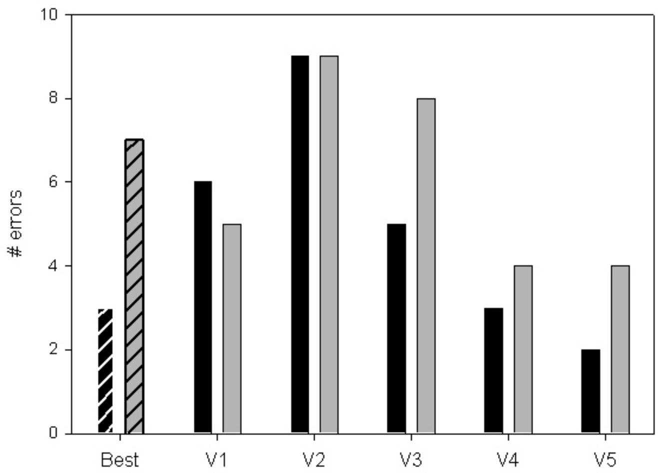

Figure 2: Number of errors made by the different ensembles (voting schemes from 1 to 5, and the performance of the best classifier on the left). Bars in black correspond to the ensemble of all the classifiers and grey bars to the ensemble of all but the best.

6 Conclusions

We have shown the performance of classifier ensembles for classifying a symbolically represented melody into a given music genre, using statistical description models. In previous works we have shown the feasibility of using these kind of data and representations to approach the problem, but by constructing an ensemble using different classifiers, their votes are "averaged" and this reduces the risk of choosing the wrong classifier.

Among all the voting schemes tested, the approaches based on scaling the weights to a range established by the best and worst classifiers have shown the best classification accuracy, which is slightly better than the most accurate individual classifier utilized. Evidence of the robustness of these best-worst scale based ensembles has also been shown. After removing the best classifier from the ensembles, they still managed to perform fairly better than any of the remaining individual classifiers.

Further work is needed to test the robustness of this scheme to other music genres, using different classification paradigms, and combination techniques, perhaps taking advantage of the capability of the combination schemes presented here to ouput membership probabilities for each genre, given a sample melody, as stated in section 4.3.2.

7 Acknowledgments

The authors would like to thank Francisco Moreno-Seco for his help, advise, and programming.

References

[1] Jianjun Zhu, Xiangyang Xue, and Hong Lu. Musical genre classification by instrumental features. In *Int. Computer Music Conference, ICMC 2004*, pages 580–583, 2004.

[2] B. Whitman, G. Flake, and S. Lawrence. Artist detection in music with minnowmatch. In *Proc. IEEE Workshop on Neural Networks for Signal Processing*, pages 559–568, 2001.

[3] G. Tzanetakis, A. Ermolinskyi, and P. Cook. Pitch histograms in audio and symbolic music information retrieval. *Journal of New Music Research*, 32(2):143–152, June 2003.

[4] P. P. Cruz, E. Vidal, and J. C. Pérez-Cortes. Musical style identification using grammatical inference: The encoding problem. In Alberto Sanfeliu and José Ruiz-Shulcloper, editors, *Proc. of CIARP 2003*, pages 375–382, La Habana, Cuba, 2003.

[5] Cory McKay and Ichiro Fujinaga. Automatic genre classification using large high-level musical feature sets. In *Int. Conf. on Music Information Retrieval, ISMIR 2004*, pages 525–530, 2004.

[6] Peter Knees, Elias Pampalk, and Gerhard Widmer. Artist classification with web-based data. In *Proceedings of the 5th International ISMIR 2004 Conference*, Barcelona, Spain, October 2004.

[7] Pedro J. Ponce de León and José M. Iñesta. Statistical description models for melody analysis and characterization. In *Proceedings of the 2004 International Computer Music Conference*, pages 149–156. International Computer Music Association, 2004.

[8] C. Pérez-Sancho, J. M. Iñesta, and J. Calera-Rubio. Style recognition through statistical event models. In *Proceedings of the Sound and Music Computing Conference, SMC '04*, 2004.

[9] T. Dietterich. Ensemble methods in machine learning. In *First Internacional Workshop on Multiple Classifier Systems*, pages 1–15. 2000.

[10] L. I. Kuncheva. That elusive diversity in classifier ensembles. In *Proc. 1st Iberian Conf. on Pattern Recognition and Image Analysis (IbPRIA'03)*, volume 2652 of *Lecture Notes in Computer Science*, pages 1126–1138. 2003.

[11] Derek Partridge and Niall Griffith. Multiple classifier systems: Software engineered, automatically modular leading to a taxonomic overview. *Pattern Analysis and Applications*, 5:180–188, 2002.

[12] Jeremy Pickens. A survey of feature selection techniques for music information retrieval. Technical report, Center for Intelligent Information Retrieval, Departament of Computer Science, University of Massachussetts, 2001.

[13] P. J. Ponce de León and J. M. Iñesta. Feature-driven recognition of music styles. In *1st Iberian Conference on Pattern Recognition and Image Analysis. LNCS, 2652*, pages 773–781, 2003.

[14] R. B. D'Agostino and M. A. Stephens. *Goodness-of-Fit Techniques*. Marcel Dekker, Inc., New York, 1986.

[15] Shyamala Doraisamy and Stefan Rüger. Robust polyphonic music retrieval with n-grams. *Journal of Intelligent Information Systems*, 21(1):53–70, 2003.

[16] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. John Wiley, 1991.

[17] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. John Wiley and Sons, 2000.

[18] Andrew McCallum and Kamal Nigam. A comparison of event models for naive bayes text classification. In *AAAI-98 Workshop on Learning for Text Categorization*, pages 41–48, 1998.

[19] P. Domingos and M. Pazzani. Beyond independence: conditions for the optimality of simple bayesian classifier. *Machine Learning*, 29:103–130, 1997.

[20] Padraig Cunningham and John Carney. Diversity versus quality in classification ensembles based on feature selection. In *Machine Learning: ECML 2000, 11th*

European Conference on Machine Learning, volume 1810 of *Lecture Notes in Computer Science*, pages 109–116. 2000.

[21] A. Blum. Empirical support for winnow and weighted-majority based algorithms: Results on a calendar scheduling domain. *Machine Learning*, 26(1):5–23, 1997.

[22] D. Opitz and J. Shavlik. Generating accurate and diverse members of a neural-network ensemble. In D. Touretzky, M. Mozer, and M. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 535–541, 1996.

[23] E. Stamatatos and G. Widmer. Music performer recognition using an ensemble of simple classifiers. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, pages 335–339, 2002.