NOVELTY DETECTION BASED ON SPECTRAL SIMILARITY OF SONGS

Arthur Flexer[1], **Elias Pampalk**[1], **Gerhard Widmer**[1,2]
[1] Austrian Research Institute for Artificial Intelligence (OFAI)
Freyung 6 /6, A-1010 Vienna, Austria
[2] Department of Computational Perception
Johannes Kepler University
Altenberger Str. 69, A-4040 Linz, Austria
{arthur,elias,gerhard}@ofai.at

ABSTRACT

We are introducing novelty detection, i.e. the automatic identification of new or unknown data not covered by the training data, to the field of music information retrieval. Two methods for novelty detection - one based solely on the similarity information and one also utilizing genre label information - are evaluated within the context of genre classification based on spectral similarity. Both are shown to perform equally well.

Keywords: novelty detection, spectral similarity, genre classification

1 INTRODUCTION

Novelty detection is the identification of new or unknown data that a machine learning system is not aware of during training (see (Markou & Singh 2003a) for a review). It is a fundamental requirement for every good machine learning system to automatically identify data from regions not covered by the training data since in this case no reasonable decision can be made. This paper is about introducing novelty detection to the field of music information retrieval where so far the problem has been ignored.

For music information retrieval, the notion of central importance is musical similarity. Proper modeling of similarity enables automatic structuring and organization of large collections of digital music, and intelligent music retrieval in such structured "music spaces". This can be utilized for numerous different applications: genre classification, play list generation, music recommendation, etc. What all these different systems lack so far is the ability to decide when a new piece of data is too dissimilar for making a decision. Let us e.g. assume the following user scenario: a user has on her hard drive a collection of songs classified into the three genres 'hip hop', 'punk' and 'death metal'; given a new song from a genre not yet covered by the collection (say, a 'reggae' song), the system should mark this song as 'novel' therefore needing manual processing instead of automatically and falsely classifying it into one of the three already existing genres (e.g. 'hip hop'). Another example is the automatic exclusion of songs from play lists because they do not fit the overall flavor of the majority of the list. Novelty detection could also be utilized to recommend new types of music different from a given collection if users are longing for a change.

We will present two methods for novelty detection based on spectral similarity of songs and evaluate them within a genre classification context (see e.g. (Aucouturier & Pachet 2004)). Spectral similarity is computed using Mel Frequency Cepstrum Coefficients (MFCC) and Gaussian Mixture Models (GMM). After introducing the data base used in the study as well as the employed preprocessing (Sec. 2), we will describe the methods of GMMs and novelty detection (Sec. 3), present our experiments and results (Sec. 4) which is followed by discussion (Sec. 5) and conclusion (Sec. 6).

Table 1: Statistics of our data set

			Artists/Genre		Tracks/Genre	
Genres	Artists	Tracks	Min	Max	Min	Max
22	103	2522	3	6	45	259

2 DATA

For our experiments we used an in-house collection containing $S = 2522$ songs belonging to $G = 22$ genres. Details are given in Tables 1 and 2. The data set has mainly been organized according to genre/artist/album. Thus, all pieces of the same artist (and album) are assigned to the same genre, which is a questionable but common practice. The genres are user defined, far from perfect and therefore quite a realistic setting: there are two different definitions of trance, there are overlaps, for example, jazz and jazz guitar, heavy metal and death metal etc.

From the 22050Hz mono audio signals two minutes from the center of each song are used for further analysis. We divide the raw audio data into overlapping frames of short duration and use Mel Frequency Cepstrum Coefficients (MFCC) to represent the spectrum of each frame. MFCCs are a perceptually meaningful and spec-

Table 2: List of genres for our data set

a cappella	acid jazz	blues
bossa nova	celtic	death metal
drum and bass	downtempo	electronic
euro-dance	folk-rock	german hip hop
hard core rap	heavy metal/thrash	italian
jazz	jazz guitar	melodic metal
punk	reggae	trance
trance2		

trally smoothed representation of audio signals. MFCCs are now a standard technique for computation of spectral similarity in music analysis (see e.g. (Logan 2000)). The frame size for computation of MFCCs for our experiments was $23.2ms$ (512 samples), with a hop-size of $11.6ms$ (256 samples) for the overlap of frames. The average energy of each frame's spectrum was subtracted. We used the first 20 MFCCs for all our experiments.

3 METHODS

3.1 Computing spectral similarity of songs

The following approach to music similarity based on spectral similarity pioneered by (Logan & Salomon 2001) and (Aucouturier & Pachet 2002) is now seen as one of the standard approaches in the field of music information retrieval. For a given music collection of S songs, each belonging to one of G music genres, it consists of the following basic steps:

- for each song, compute MFCCs for short overlapping frames as described in Sec. 2
- train a Gaussian Mixture Model (GMM) for each of the songs
- compute a similarity matrix between all songs using the likelihood of a song given a GMM
- based on the genre information, do nearest neighbor classification using the similarity matrix

The last step of genre classification can be seen as a form of evaluation. Since usually no ground truth with respect to music similarity exists, each song is labeled as belonging to a music genre using e.g. music expert advice. High genre classification results are taken to indicate good similarity measures. The winning entry to the ISMIR 2004 genre classification contest[1] by Elias Pampalk followed basically the above described approach.

A Gaussian Mixture Model (GMM) models the density of the input data by a mixture model of the form

$$p(x) = \sum_{m=1}^{M} P_m \mathcal{N}[x, \mu_m, U_m] \quad (1)$$

where P_m is the mixture coefficient for the m-th component, $\mathcal{N}$ is the normal density and μ_m and U_m are the mean vector and covariance matrix of the m-th mixture. The log-likelihood function is given by

$$L(X) = \frac{1}{T} \sum_{t=1}^{T} log(p(x_t)) \quad (2)$$

for a data set X containing T data points. This function is maximized both with respect to the mixing coefficients P_m and with respect to the parameters of the Gaussian basis functions using Expectation-Maximization (see e.g. (Bishop 1995)). For all our experiments we used $M = 30$ components. To compute similarity between two songs A and B, we sample 400 points S^A from model A and compute the log-likelihood of these samples given model B using Equ. 2 which gives $L(S^A|B)$. Reversing the roles of A and B we get $L(S^B|A)$. Summing these two log-likelihoods and subtracting the self-similarity for normalization yields the following similarity function:

$$d(A,B) = L(S^A|B) + L(S^B|A) - L(S^A|A) - L(S^B|B) \quad (3)$$

3.2 Algorithms for novelty detection

Ratio-reject: The first reject rule is based on density information about the training data captured in the similarity matrix. An indication of the local densities can be gained from comparing the distance between a test object X and its nearest neighbor in the training set $NN^{tr}(X)$, and the distance between this $NN^{tr}(X)$ and its nearest neighbor in the training set $NN^{tr}(NN^{tr}(X))$ (Tax & Duin 1998). The object is regarded as novel if the first distance is much larger than the second distance. Using the following ratio

$$\rho(X) = \frac{\|d(X, NN^{tr}(X))\|}{\|d(NN^{tr}(X), NN^{tr}(NN^{tr}(X)))\|} \quad (4)$$

we reject X if:

$$\rho(X) > \varepsilon[\rho(X^{tr})] + s * std(\rho(X^{tr})) \quad (5)$$

with $\varepsilon[\rho(X^{tr})]$ being the mean of all quotients $\rho(X^{tr})$ inside the training set and $std(\rho(X^{tr}))$ the corresponding standard deviation (i.e. we assume that the $\rho(X^{tr})$ have a normal distribution). Parameter s can be used to change the probability threshold for rejection. Setting $s = 3$ means that we reject a new object X if its ratio $\rho(X)$ is larger then the mean ρ within the training set plus three times the corresponding standard deviation. In this case a new object is rejected because the probability of its distance ratio $\rho(X)$ is less than 1% when compared to the distribution of $\rho(X^{tr})$. Setting $s = 2$ rejects objects less probable than 5%, $s = 1$ less than 32%, etc.

Knn-reject: It is possible to directly use nearest neighbor classification to reject new data with higher risk of being misclassified (Hellman 1970):

reject X if **not**:

$$g(NN1^{tr}(X)) = g(NN2^{tr}(X)) = \ldots = g(NNk^{tr}(X)) \quad (6)$$

[1] ISMIR 2004, 5th Intern. Conf. on Music Information Retrieval, Spain, 2004; see `http://ismir2004.ismir.net/ISMIR-Contest.html`

with $NNi^{tr}(X))$ being the ith nearest neighbor of X in the training set, $g()$ a function which gives the genre information for a song and $i = 1, \ldots, k$. A new object X is rejected if the k nearest neighbors do not agree on its classification. This approach will work for novelty detection if new objects X induce high confusion in the classifier. The higher the value for k the more objects will be rejected.

4 RESULTS

To evaluate the two novelty detection approaches described in Sec. 3.2 we use the following approach shown as pseudo-code in Table 3. First we set aside all songs belonging to a genre `g` as new songs (`[new,data]=separate(alldata,g)`) which yields data sets `new` and `data` (all songs not belonging to genre g). Then we do a ten-fold cross-validation using `data` and `new`: we randomly split `data` into `train` and `test` fold (`[train,test] = split(data,c)`) with `train` always consisting of 90% and `test` of 10% of `data`. We compute the percentage of `new` songs which are rejected as being novel (`novel_reject(g,c) = novel(new)`) and do the same for the `test` songs (`test_reject(g,c) = novel(test)`). Last we compute the accuracy of the nearest neighbor classification on `test` data that has not been rejected as being novel (`accuracy(g,c) = classify(test(not test_reject))`). The evaluation procedure gives $G \times C$ (22×10) matrices of `novel_reject`, `test_reject` and `accuracy` for each parameterization of the novelty detection approaches.

Table 3: Outline of Evaluation Procedure

```
for g = 1 : G
  [new,data] = separate(alldata,g)
  for c = 1 : 10
    [train,test] = split(data,c)
    novel_reject(g,c) = novel(new)
    test_reject(g,c) = novel(test)
    accuracy(g,c) =
      classify(test(not test_reject))
  end
end
```

The results for novelty detection based on the Ratio-reject and the Knn-reject rule are given in Figs. 1 and 2 as Receiver Operating Characteristic (ROC) curves (Metz 1978). To obtain an ROC curve the fraction of false positives (object is not novel but it is rejected, in our case `test_reject`) is plotted versus the fraction of true positives (object is novel and correctly rejected, in our case `novel_reject`). An ROC curve shows the tradeoff between how sensitive and how specific a method is. Any increase in sensitivity will be accompanied by a decrease in specificity. If a method becomes more sensitive towards novel objects it will reject more of them but at the same it will also become less specific and also falsely reject more non-novel objects. Consequently, the closer a curve follows the left-hand border and then the top border of the ROC space, the more accurate the method is. The closer the curve comes to the 45-degree diagonal of the ROC space, the less accurate the method. We plot the mean `test_reject` versus the mean `novel_reject` for falling numbers of s (Ratio-reject) and growing numbers of k (Knn-reject). In addition the mean `accuracy` for each of the different values of s and k are depicted as separate curves. All means are computed across all 22×10 corresponding values. The accuracy without any rejection due to novelty detection is 70%.

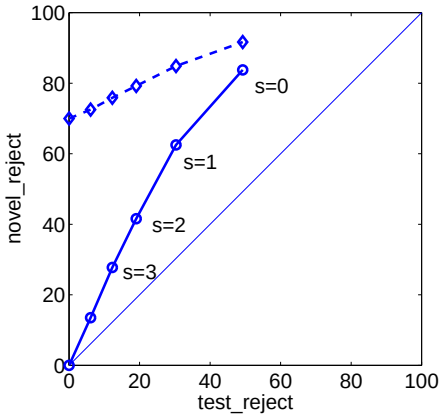

Figure 1: Ratio-reject ROC, mean `test_reject` vs. `novel_reject` (circles, solid line) and `accuracy` (diamonds, broken line) for 'no rejection', s=5,3,2,1,0.

Ratio-reject: The results for novelty detection based on the Ratio-reject rule are given in Fig. 1. With the probability threshold for rejection set to $s = 2$ (rejection because data is less probable than 5%), the accuracy rises up to 79% while 19% of the `test` songs are falsely rejected as being novel and therefore not classified at all and 42% of the `new` songs are being rejected correctly. If one is willing to lower the threshold to $s = 0$ (rejection because data is less probable than 50%) the accuracy is at 92% with already 49% of the `test` songs rejected erroneously and 84% of the `new` songs rejected correctly.

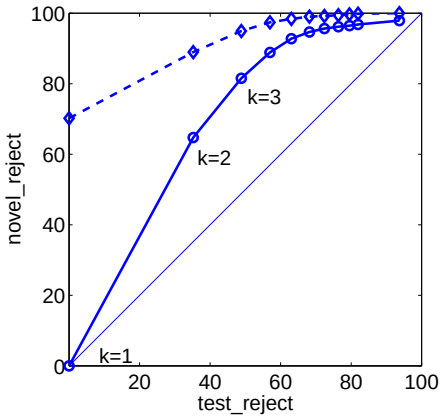

Figure 2: Knn-reject ROC, mean `test_reject` vs. `novel_reject` (circles, solid line) and `accuracy` (diamonds, broken line) for k=1 (no rejection) and k=2,3,4,5,6,7,8,9,10,20.

Knn-reject: The results for novelty detection based on the Knn-reject rule are given in Fig. 2. If k is set to 2 the accuracy rises up to 89% while 35% of the `test` songs are wrongly rejected as being novel and therefore not classified at all and 65% of the `new` songs are being rejected correctly. With $k = 3$ the accuracy values start to saturate at 95% with already 49% of the `test` songs rejected erroneously and 81% of the `new` songs rejected correctly.

5 DISCUSSION

We have presented two approaches to novelty detection, where the first (Ratio-reject) is based directly on the distance matrix and does not, contrary to Knn-reject, need the genre labels. When comparing the two ROC curves given in Figs. 1 and 2 it can be seen that both approaches work approximately equally well. E.g. the performance of the Ratio-reject rule with $s = 1$ resembles that of the Knn-reject rule with $k = 2$. The same holds for $s = 0$ and $k = 3$. Also the increase in accuracy is comparable for both methods. Depending on how much specificity one is willing to sacrifice, the accuracy can be increased from 70% to well above 90%. Looking at both ROC curves, we would like to state that they indicate quite fair accurateness of both novelty detection methods.

When judging genre classification results, it is important to remember that the human error in classifying some of the songs gives rise to a certain percentage of misclassification already. Inter-rater reliability between a number of music experts is usually far from perfect for genre classification. Given that the genres for our data set are user and not expert defined and therefore even more problematic (see Sec. 2), it is not surprising that there is a considerable decrease in specificity for both methods.

Of course there is still room for improvement in novelty detection for music similarity. The two presented methods are a first attempt to tackle the problem and could probably be improved themselves. One could change the Knn-reject rule given in Equ. 6 by introducing a weighting scheme which puts more emphasis on closer than on distant neighbors. Then there is a whole range of alternative methods which could be explored: probabilistic approaches (see e.g. (Bishop 1994)), Bayesian methods (MacKay 1992) and neural network based techniques (see (Markou & Singh 2003b) for an overview).

Finally we would like to comment that whereas the Knn-reject rule is bound to the genre classification framework, Ratio-reject is not. Knn-reject probably is the method of choice if classification is the main interest. Any algorithm that is able to find a range of nearest neighbors in a data base of songs can be used together with the Knn-reject rule. Ratio-reject on the other hand has an even wider applicability. It is a general method to detect novel songs given a similarity matrix of songs. Since it does not need genre information it could be used for anything from play list generation and music recommendation to music organization and visualization.

6 CONCLUSION

We introduced novelty detection, i.e. the automatic identification of new or unknown data not covered by the training data, to the field of music information retrieval. We presented two different methods for novelty detection with the first relying solely on the similarity information and the second also utilizing genre label information. Both have been shown to perform equally well in terms of sensitivity, specificity and accuracy within a genre classification context. We also discussed the potential of novelty detection to improve a wide range of music information retrieval applications.

ACKNOWLEDGEMENTS

Parts of the MA Toolbox (Pampalk 2004) and the Netlab Toolbox[2] have been used for this work. This research was supported by the EU project FP6-507142 SIMAC[3]. The Austrian Research Institute for Artificial Intelligence is supported by the Austrian Federal Ministry of Education, Science and Culture and the Austrian Federal Ministry for Transport, Innovation and Technology.

References

[Aucouturier & Pachet 2002] Aucouturier J.-J., Pachet F.: Music Similarity Measures: What's the Use?, in Proc. of the 3rd Intern. Conf. on Music Information Retrieval (ISMIR'02), pp. 157-163, 2002.

[Aucouturier & Pachet 2004] Aucouturier, J.-J., Pachet F.: Improving Timbre Similarity: How high is the sky? Journal of Negative Results in Speech and Audio Sciences, 1(1), 2004.

[Bishop 1994] Bishop C.: Novelty detection and neural network validation, Proceedings of the IEE Conference on Vision and Image Signal Processing, pp.217-222, 1994.

[Bishop 1995] Bishop C.M.: Neural Networks for Pattern Recognition, Clarendon Press, Oxford, 1995.

[Hellman 1970] Hellman M.E.: The nearest neighbour classification with a reject option, IEEE Transaction on Systems Science and Cybernetics, Vol. 6, No. 3, pp.179-185, 1970.

[Logan 2000] Logan B.: Mel Frequency Cepstral Coefficients for Music Modeling, Proc. of the International Symposium on Music Information Retrieval (ISMIR'00), 2000.

[Logan & Salomon 2001] Logan B., Salomon A.: A music similarity function based on signal analysis, IEEE International Conf. on Multimedia and Expo, Tokio, Japan, 2001.

[Markou & Singh 2003a] Markou M., Singh S.: Novelty detection: a review-part 1: statistical approaches, Signal Processing, 83(12):2481-2497, 2003.

[Markou & Singh 2003b] Markou M., Singh S.: Novelty detection: a review-part 1: neural network based approaches, Signal Processing, 83(12):2499 - 2521, 2003.

[MacKay 1992] MacKay D.J.C.: The evidence framework applied to classification networks, Neural Computation, 4:720-736, 1992.

[Metz 1978] Metz C.E.: Basic principles of ROC analysis, Semin Nucl Med, 8(4):283-98, 1978.

[Pampalk 2004] Pampalk E.: A Matlab Toolbox to compute music similarity from audio, in Proceedings of the 5th International Conference on Music Information Retrieval (ISMIR'04), Barcelona, Spain, pp.254-257,2004.

[Tax & Duin 1998] Tax D.M.J., Duin R.P.W.: Outlier detection using classifier instability, in Amin A. et al. (eds.), Advances in Pattern Recognition, Proc. Jont IAPR Int. Workshop SSPR'98 and SPR'98, Lecture Notes in Computer Science, Springer, pp. 593-601, 1998.

[2] `http://www.ncrg.aston.ac.uk/netlab`

[3] `http://www.semanticaudio.org`