RECOGNITION OF MUSIC TYPES

Hagen Soltau, Tanja Schultz, Martin Westphal, Alex Waibel

Interactive Systems Laboratories
University of Karlsruhe (Germany), Carnegie Mellon University (USA)
{soltau,tanja,westphal,waibel}@ira.uka.de

ABSTRACT

This paper describes a music type recognition system that can be used to index and search in multimedia databases. A new approach to temporal structure modeling is supposed. The so called ETM-NN (Explicit Time Modelling with Neural Network) method uses abstraction of acoustical events to the hidden units of a neural network. This new set of abstract features representing temporal structures, can be then learned via a traditional neural networks to discriminate between different types of music. The experiments show that this method outperforms HMMs significantly.

1. INTRODUCTION

As the demand for multimedia databases grows, the development of information retrieval systems including information about music is of increasing concern. Radio stations and music TV channels hold archives of millions of music tapes and video clips. For an easy and fast search in databases automatically indexing of tapes and clips is required. For this purpose the music type is obviously one important information. A music type recognizer would enable an intelligent car radio system to automatically select favored music channels.

In this paper a recognition system is described that discriminate four major categories of music types: Rock, Pop, Techno and Classic. Bands like *ZZ-Top* or *Metallica* are assigned to the category Rock. Softer music like *Sade* belongs to Pop music. Hard and fast beats are typical characteristics of Techno. *Haydn*, *Mussogorsky* or *Mozart* and alike are representatives for classical music.

A music type recognizer must cope with temporal structures of acoustic signals. These of speech signals are often modeled with HMMs [4] but a drawback is their poor discriminative power. In the contrary neural networks provide a very good discrimination but to compete with temporal structures special topologies are needed like those proposed by Elman and Jordan [2]. Since temporal information is stored in context units, such networks have to learn temporal structure from the context units.

We suggest a new idea to represent temporal structures of input signal in order to better compete with temporal structure variations in the inputs. We call this new architecture ETM-NN (Explicit Time Modeling with Neural Networks). ETM-NN uses statistical analysis of temporal structure to provide some new features to the whole network.

The paper is organized as follows: In the next section we give an overview of the ETM-NN. Section 3 describes the experimental setup, the database and the preprocessing step. Results on our experiments are reported and discussed in Section 4.

2. EXPLICIT TIME MODELING WITH NEURAL NETWORKS

2.1. Overview

From an acoustic point of view, music can be described as a sequence of acoustic events. For the music type recognition it is relevant to extract information about the temporal structure of this sequences. Therefore, we have to transform the acoustic signal into a sequence of abstract acoustic events. This transformation is described in the next subsection. After this, statistical parameters will be derived from the sequences. Frequencies of events and transitions in the sequence are calculated. The derived parameters are combined into one vector which contains the temporal structure information of the sequence. This so-called characteristic vector of a piece of music is the input of a 3-layer feed-forward network which is trained to recognize music types. The overall system structure is depicted in figure 1.

2.2. Learning acoustic events

Speech events are described in terms of phones. Speech recognizers use often phonemes or similar units for the acoustic modeling. But how to describe the acoustic events of music? Modeling for example by notes would have many disadvantages: Often notes are played simultaneously (accords, polyphonic music) and music samples contain additional voices and other sounds. For example, one can hear raindrops falling in some exotic pieces of Techno.

Thus it is difficult to extract single notes from the signal and works on simple monophonic music. We suggest to

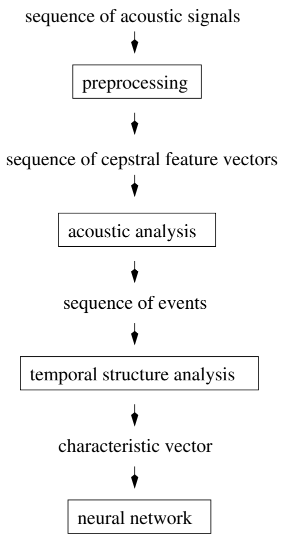

Figure 1: System structure for explicit time modeling with neural networks

learn the relevant acoustic events in an unsupervised manner.

For an autoassociative trained 3-layer feed-forward network (encoder nets) it is often observed that the hidden layer represents approximately the principle components. In [7] it is shown that the first half of the feed-forward network performs a specific nonlinear transformation of the input data into a space in which the discrimination should be simpler.

In fact, the activation of the hidden units are a compact representation of the input feature vector, but our aim is a good representation to recognize music types instead of autoassociate the input vector itself. Therefore, we train the network to associate the input signal to the music type. The purpose of this network is not to get a perfect music type recognizer but to learn abstract acoustic events in the hidden layer. Input of the network are the $10 * 5$ cepstral coefficients. The hidden layer has 10 units. After the network is trained we only use the hidden units, the output units are ignored.

The abstract event e_i occures if the hidden unit i of this trained network has the highest activation of all hidden units. The score of the event e_i corresponds to the activation of hidden unit i. Obviously, the number of possible events is the same as the number of hidden units. Therefore we use ten events in our system. The minimum duration of an event is the context size of the input feature vector (0.4 seconds).

To transform the acoustic signal into a sequence of such events we put each input feature vector of the signal into the network and compute the hidden activation. With this procedure, we get the corresponding abstract event for each input vector. Figure 2 illustrates the activation of the hidden units for each input vector of a sample of a Techno piece "Happy Rave". At the beginning of the sample hidden unit

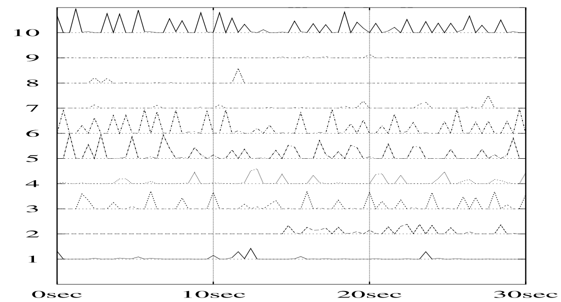

Figure 2: Activation of the 10 hidden units for a sample of techno "Happy Rave"

10 has the highest activation, followed by hidden units 6, 5, 10, 3, 5, 6, 5, 10, 6, 10 and so on. The corresponding event sequence is e_{10}, e_6, e_5, e_{10}, e_3, e_5, e_6, e_5, e_{10}, e_6, e_{10} and so forth.

We require for our definition of events that hidden units are not activated simultaneously, because we assign an event to the highest activated unit. A solution to make sure that only one hidden unit is active would be to perform a competive learning rule in the hidden layer during training the network, but we observed that this is not needed. An analysis of the hidden activation has shown that the rate of simultaneously activated hidden units is small.

2.3. Analysis of temporal structures

In the next step temporal structure information are extracted from the event sequence. This is done by the temporal structure analysis module. Input of this module is the event sequence obtained by the procedure described above. The analysis module produces the characteristic vector of a piece of music by counting, how often events e_i, event pairs $e_i e_j$ or event triplets $e_i e_j e_k$ occur in the sequence. For example, the piece of Techno depicted in figure 2 has many transitions from e_5 to e_6. The counts of the events e_1 and e_9 will be zero, these events are not active over the whole sample. We call these features in analogy to statistical language models unigrams, bigrams and trigrams, but our intention is to use the counts as features instead as probabilities.

Additionally the average event duration is computed. Duration means, how long a hidden unit has the highest activation. The difference to the unigram $c(e_i)$ is that the unigram counts the occurencies but not their durations. Also we use features to represent the scores of the event activation. The analysis module computes maximum, mean and variance of the activation score for each event.

To summarize the following features are computed:

- unigram-counts $c(e_i)$
- bigram-counts $c(e_i, e_j)$

- trigram-counts $c(e_i, e_j, e_k)$
- event durations
- mean, maximum, variance of event activation

The features are normalized over the length of the sequence and combined into a single vector that we call characteristic vector. Due to the high number of potential n-grams the dimension of the vector is quite big. Therefore it is necessary to reduce the dimension. We select the n best features acording to their disriminative power. The reduction results in an easy and fast learning of the final neural network.

2.4. Recognition engine

As a final step, we trained a neural network to recognize the music type. Input of the network is the characteristic vector obtained by the temporal structure analysis module. The temporal structure information is represented in the reduced characteristic vector. Therefore the network has not to deal with a sequence of feature vectors anymore.

3. EXPERIMENTAL SETUP

We carried out two kinds of experiments. We evaluated a standard approach using HMMs to model the music types. The HMM approach is compared with the ETM-NN training scheme. All reported results are measured on the evaluation set from the database described in the next subsection. We used the same preprocesing step for both systems.

3.1. Database

The database consists of 3 hours of audio data for the four categories Rock, Pop, Techno and Classic. We collected 360 samples of music of approximately 30 seconds. Each music type has an equal number of samples in the database. To avoid a specialization to bands we restrict the maximum number of music pieces for each band to six. The database is divided in three artist-disjunct sets for training (67%), cross validation (13%) and evaluation (20%). The samples were originally obtained from compact discs and sampled down to 16 kHz. Additionally, we merged both channels of the stereo signal into one mono signal.

3.2. Preprocessing

The instruments and their timbre are characteristic for different music types. For example, a piano is more typical for classic and, on the other hand, an electrical guitar is more typical for Rock music.

The sound of an instrument consists of two components, the excitation (fundamental tone) and the resonance characteristics (timbre). Physically, timbre can be expressed as the relation between the energy of the partials (overtones). Therefore, we compute a short-time power spectrum with a window size of 50 ms every 40 ms. To filter out the fundamental frequency we compute only the first 5 cepstral coefficients of each window. The resulting feature vector contains context of 10 adjacent frames so that the dimension of the feature vector is $10 * 5 = 50$. Such a feature vector provides information for $10 * 40$ ms= 0.4 seconds of the signal.

4. RESULTS AND DISCUSSION

4.1. HMM results

As for the ETM-NN we used cepstral coefficients as input for the HMM system. The system consists of four HMMs corresponding to the four music types. Each HMM has four states. We investigated different transition models. Figure 3 shows on the left side a left-to-right model with one back loop on the other side and an ergodic model.

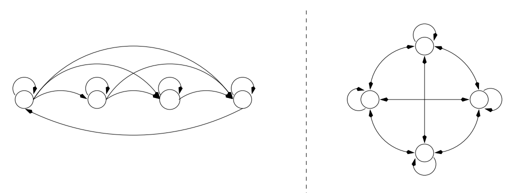

Figure 3: left-to-right with one back loop and ergodic HMM transition models

In both HMM topologies we use a mixture of three gaussians to parameterize the emission probabilities. The bootstrap mechanism starts with one gaussian. We write labels after five full forward-backward iterations. These labels are used for initialization of the mixture of gaussians followed by ten further training iterations. The transition probabilities are also trained. The HMM recognizer with the ergodic transiton model achieves 76.4% recognition rate as shown in table 1. In contrary the HMM with the restricted transition model has a better performance 79.2%. A reason for that is that the reestimation of transition probabilities would be more unreliable if more potential state transitions are existing.

4.2. ETM-NN results

We investigated different types of characteristic vectors. First, we restricted the temporal structure information to event unigrams, durations and activation scores. Bigrams and trigrams were excluded. The performance of the ETM-NN was 81.9%. In the second experiment we added bigram information to the characteristic vector. This step increased the recognition rate to 84.7%. If we add also trigram features the performance can be improved to 86.1%. Obviously the bigram and trigram features contain useful information about the temporal structure.

System	recognition rate
HMM with:	
left-to-right transitions	79.2%
ergodic transitions	76.4%
ETM-NN with:	
unigrams, duration, activation	81.9%
+ bigrams	84.7%
+ trigrams	86.1%

Table 1: Performance of HMM and ETM-NN

Detailed results for each music type are shown in table 2. The differences between the recognition performance across the music types are interesting. It seems that samples of Techno and Classic are easy to discriminate. The recognition of samples of Rock and Pop seems to be more difficult. This results are conform with human perception. We carried out a perceptual study with a group of 37 test subjects. The subjects were exposed to the same samples that we used for the test set. They are asked to classify the music type. The results of this experiment are similar to the results obtained with our recognition system. Human confusions in this experiment are similar to confusions of the ETM-NN system. Details of this perceptual study can be found in [6].

s/r[1]	Rock	Pop	Techno	Classic
Rock	72.2%	27.3%	0%	0%
Pop	11.1%	83.3%	5.6%	0%
Techno	0%	0%	94.4%	5.6%
Classic	0%	5.6%	0%	94.4%

Table 2: Confusions obtained by the ETM-NN with trigrams

We have modified the ETM-NN to an online system by computing characteristic vectors on subsequences of events. Figure 4 shows the recognition rate as a function of duration for each music type separtly. The system needs only a few seconds to recognize pieces of Techno or Classic. The recognition of Rock samples needs much more time and is more difficult. The temporal progress of the recognition of Pop music shows also that the recognition rate can be decreased for a short period through non typical acoustics.

5. CONCLUSIONS

We presented a music type recognizer for the types Rock, Pop, Techno and Classic which achieves a recognition rate

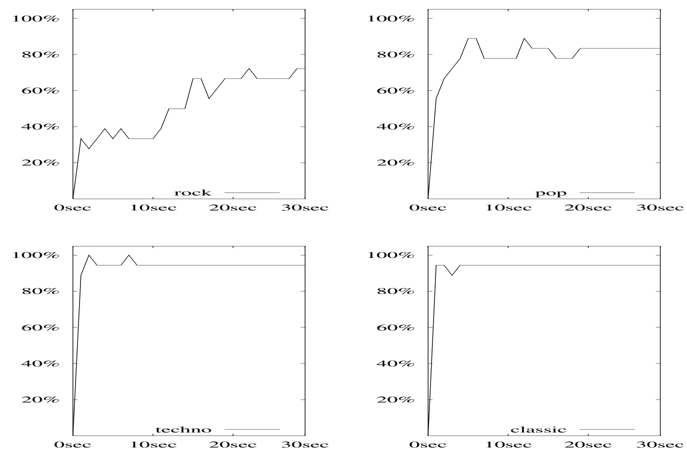

Figure 4: recogniton rate as a function of duration for each music type

of 86.1%. Our ETM-NN approach combines discriminative power of neural networks with a direct modeling of temporal structures. Further research will explore the use of direct modeling of temporal structures for acoustic signals.

6. ACKNOWLEDGEMENTS

The authors wish to thank all members of the Interactive Systems Labs, especially Thomas Schaaf, for useful discussions and active support.

7. REFERENCES

[1] S. Dixon: *Multiphonic Note Identification*, Proc. of the 19th Austral. Conference, Melborne, 1996.

[2] J.L. Elman: *Finding structures in time*, Cognitive Science, Vol. 14, 1990.

[3] P. Gallinari, S. Thiria, F. Badran, F. Fogelman-Soulie: *On the relations between discriminant analysis and multilayer perceptron*, Neural Networks, Vol. 4, 1991.

[4] L.R. Rabiner, B.H. Juang: *An introduction to hidden Markov Models*, IEEE ASSP Magazine, Jan. 1986.

[5] J. Saunders: *Real-Time Discrimination of Broadcast Speech-Music*, Proc. ICASSP 1996

[6] H. Soltau: *Erkennung von Musikstilen*, Diplomarbeit, Universität Karlsruhe, 1997.

[7] A. Webb, D. Loewe : *The optimized internal representation of multilayer classifier networks performs non-linear discriminant analysis*, Neural Networks, Vol. 3, 1990.

[1] stimulus/response: entry of row i column j is the rate how many samples from type i was classified as type j