MediaEval 2017 AcousticBrainz Genre Task: Multilayer Perceptron Approach

Khaled Koutini, Alina Imenina, Matthias Dorfer, Alexander Rudolf Gruber, Markus Schedl
Johannes Kepler University Linz, Austria
khaled.koutini@jku.at,markus.schedl@jku.at

ABSTRACT

This report describes the approach developed by the JKU team for the MediaEval 2017 AcousticBrainz Genre Task. After experimenting with various classi□ers on the development dataset, our □nal approach is based on multilayer perceptron classi□ers.

1 INTRODUCTION

We present an approach for recognizing genre for unknown music recordings given the data provided in the AcousicBrainz dataset [5]. Details about data, task, and evaluation are described in [2].
Our work is developed for both subtasks of the MediaEval 2017 AcousticBrainz Genre Task. For the single-source classi□cation subtask a multilayer perceptron is applied on each source. For the multiple-source classi□cation subtask we use similarity measures between sources to adjust the probability of the record belonging to a certain genre in each source.

2 APPROACH

We split the ground truth of each source using the script provided by the organizers, into a training and a validation set, where each comprises 80% and 20% of the original data respectively. The split also ensures that no recording from the same recordings group appears in both the training and validation sets, in order to avoid the album e□ect.

2.1 Features Selection

As stated in the overview paper [2], we are given for each recording a set of features extracted using *Essentia* [3]. Given the large number of provided features, a □ne-grained manual inspection for individual features is not feasible. Instead, we pick broad features groups high in the Essentia feature groups hierarchy. Namely, We use all the low level features, rhythm features except beats_count and beats_position, tonal features except chords_key, chords_scale, key_key and key_scale. Overall, this yields 2646 numerical features per recording.

2.2 Neural Network

We tried various neural network architectures and compared their performance based on the mean label-wise F-score of batches, using the Lastfm dataset. The best performing architecture is outlined in Table 1.

2.2.1 Input layer. As stated in Section 2.1, there are 2646 input features. We normalized the input using z-score normalization.

Table 1: Model Specifications. ReLU: Rectified Linear Unit [4], σ: the number of output units, which is the number of possible labels (unique genres + unique sub-genres). For training a constant batch size of 500 samples is used, and a learning rate of 0.001

Input: 2646
First layer: 4000 Dense(ReLU) + Drop-Out(0.5) 4000 Dense(tanh) + Drop-Out(0.5) 4000 Dense(sigmoid) + Drop-Out(0.5)
Second layer: Concat layer 8000 Dense + Drop-Out(0.6) Batch-Normalization layer Non-linearity (ReLU)
Output layer: k-bins sigmoid

2.2.2 First hidden layer. The □rst hidden layer is a dense layer consisting of 12000 units where the □rst 4000 units have a recti□ed linear [4] activation function, the next 4000 units have a $tanh$ activation function and the last 4000 units have a $sigmoid$ activation function. As shown in Table 1, each group of units is followed by a dropout layer with a dropout-probability of 0.5.

2.2.3 Second hidden layer. The second hidden layer consists of 8000 batch-normalized recti□ed linear units. As input to this layer we concatenate the output of the 3 groups of the □rst layer and add the second layer with no activation function or bias. We again apply dropout with a probability of 0.6.

2.2.4 Output layer. The output layer consists of k units, where k is source speci□c, denoting the number of labels of the source (genre or sub-genre), the activation function of the output layer is $sigmoid$.

2.2.5 Loss function. We used mean binary cross-entropy as loss function for the network.

2.3 Adjusting Threshold

The output of our neural network are k numerical values for each recording, as stated in Section 2.2.4. Each output is in the range $[0, 1]$ representing the probability of the label (genre or sub-genre) corresponding to the respective output neuron. If the probability of a label for a given recording is larger than a prede□ned threshold, we assign that label to the recording. Based on our experiments, we found that using a threshold of 0.5 for all of the labels results

MediaEval'17, 13-15 September 2017, Dublin, Ireland

in high precision but low recall. Since the goal of the task is to optimize precision, recall and F-score, we adjusted the threshold for each label individually to obtain the best value for these evaluation measures. Best results are obtained when using static thresholds of either 0.2 or 0.3 for all labels or by using a dynamic threshold for each label, estimated by maximizing the mean F-score.

Table 2: Lastfm validation set evaluation results. P: Precision, R: Recall, F: F-score

Average per		Threshold		
		0.2	0.3	dynamic
track (all labels)	P	0.54	0.60	0.59
	R	0.64	0.59	0.59
	F	0.55	0.55	0.54
track (genre labels)	P	0.69	0.71	0.70
	R	0.79	0.76	0.73
	F	0.71	0.71	0.70
label (all labels)	P	0.39	0.46	0.44
	R	0.36	0.32	0.35
	F	0.35	0.35	0.37
label (genre labels)	P	0.51	0.58	0.60
	R	0.58	0.53	0.53
	F	0.54	0.54	0.56

Table 2 shows the evaluation results for the mentioned threshold setups on the Lastfm validation set, using the evaluation scripts provided by the task organizers [2].

2.4 Combining Different Sources

The second part of the task [2] consisted of combining information from multiple source to predict labels of one source. To achieve this, we calculate the similarity between every label of one source and every label of all other sources, in order to adjust the probability of assigning a source label to a recording using other source's labels probabilities from models trained on these other sources. As stated in the overview paper [2], the datasets of different sources intersect. We exploit this intersection to estimate the similarity between the labels of different sources.

Labels are modeled as vectors, where each label is a vector of the recordings annotated with it in the ground truth. The similarity between labels from different sources is measured as the cosine similarity between these label vectors. Based on that we compute similarity matrices $M_{i,j}$ between different sources where element (i, j) holds the similarity of label i of the first source and label j in the second source. We use these pairwise similarities as conversion matrices to project probabilities produced by a model trained on one source to the labels of another source. For a specific recording, the probabilities P_i of source labels i are a vector of length n_i. This vector is produced by a model trained on the training set of source i. To also make use of the models trained on other sources we compute $P_j \cdot M_{j,i}$ which is a vector of the same length n_i also representing the probabilities of source i labels. However, this vector is produced by a model trained on source j by projecting the probabilities P_j using the respective conversion matrix. The final label probabilities (task 2) for a specific recording of source k are the weighted average label probabilities produced by the model trained on the recording's source training set as well as the projected label probabilities of all other sources (see Equation (1)).

$$Y_k = \frac{P_k + \frac{1}{3}\sum_{i \neq k} P_i \cdot M_{i,k}}{2} \quad (1)$$

3 RESULTS AND ANALYSIS

Table 3 shows the evaluation results on the validation sets of different sources using models trained on the respected training sets and using dynamic thresholds, as produced by the evaluation scripts provided by the task organizers [2]. We can clearly observe that

Table 3: Validation set evaluation results of the 4 sources using dynamic threshold (section 2.3). P: Precision, R: Recall, F: F-score

Average per		Source			
		Allmusic	Tagtraum	Lastfm	Discogs
track (all labels)	P	0.53	0.53	0.59	0.50
	R	0.56	0.59	0.59	0.62
	F	0.48	0.53	0.54	0.51
track (genre labels)	P	0.71	0.72	0.70	0.78
	R	0.74	0.76	0.73	0.80
	F	0.70	0.72	0.70	0.76
label (all labels)	P	0.49	0.32	0.44	0.28
	R	0.36	0.32	0.35	0.29
	F	0.40	0.30	0.37	0.27
label (genre labels)	P	0.56	0.55	0.60	0.59
	R	0.48	0.51	0.53	0.54
	F	0.51	0.52	0.56	0.56

predicting sub-genres labels is harder than predicting genre labels which might be a result of the fewer training examples of those sub-genres in the dataset.

We submitted 3 runs for the first task [2], two runs using static threshold of 0.2 and 0.3, and a run using dynamic thresholds as described in section 2.3. We also submitted 5 runs for the second task [2], two runs identical to the static threshold runs of task1, and 3 runs based on probabilities calculated as described in section 2.4 using static threshold of 0.2 and 0.3 and dynamic thresholds.

Table 4 summarizes the f-score official results [1] of our best run of each source.

Table 4: Official results [1] (F-score)

Average per	Source			
	Allmusic	Tagtraum	Lastfm	Discogs
track (all labels)	0.43	0.53	0.55	0.51
track (genre labels)	0.67	0.73	0.72	0.77
label (all labels)	0.20	0.28	0.35	0.25
label (genre labels)	0.41	0.50	0.55	0.56

REFERENCES

[1] D Bogdanov, A Porter, J Urbano, and H Schreiber. 2017. O□cial results of the MediaEval 2017 AcousticBrainz Genre Task. https://multimediaeval.github.io/2017-AcousticBrainz-Genre-Task/results/. (2017). [Online; accessed 09-September-2017].

[2] D Bogdanov, A Porter, J Urbano, and H Schreiber. 2017. The MediaEval 2017 AcousticBrainz Genre Task: Content-based Music Genre Recognition from Multiple Sources. In Working Notes Proceedings of the MediaEval 2016 Workshop. Dublin, Ireland.

[3] D. Bogdanov, N. Wack, E. Gómez, S. Gulati, P. Herrera, O. Mayor, G. Roma, J. Salamon, J.R. Zapata, and X. Serra. 2013. Essentia: An Audio Analysis Library for Music Information Retrieval. In *International Society for Music Information Retrieval (ISMIR'13) Conference*. Curitiba, Brazil, 493–498.

[4] Vinod Nair and Geo□rey E Hinton. 2010. Recti□ed linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*. 807–814.

[5] A. Porter, D. Bogdanov, R. Kaye, R. Tsukanov, and X. Serra. 2015. Acousticbrainz: a community platform for gathering music information obtained from audio. In Proceedings of the 16th International Society for Music Information Retrieval Conference. Malaga, Spain, 786–792.