MIREX 2018: AUTOMATIC DRUM TRANSCRIPTION WITH CONVOLUTIONAL NEURAL NETWORKS

Céline Jacques
IRCAM
celine.jacques@ircam.fr

Achille Aknin
IRCAM
achille.aknin@ens.fr

Axel Roebel
IRCAM
axel.roebel@ircam.fr

ABSTRACT

This extended abstract accompanies a complete submission to the 2018 MIREX drum transcription task.

1. INTRODUCTION

The automatic drum transcription system described in [1] is originally based on data representation from [2] and uses Convolutional Neural Networks (CNN) to provide an onset detection function.

The all four submissions are based on the drum transcription system from [1]. The differences of the submission are detailed in section 3.

2. TRAINING

We split the database of sound examples provided by Mirex into a training and testing sets. The testing set respects as possible the distribution of the drum instruments in the subsets of the database.

3. SUBMISSIONS

3.1 JAR5

This network is most similar to the one presented in [1]. Modifications are performed in the number of units per layer. The network is modified to get 20 filters on layer 1, 40 on layer 2 and 512 for the dense layer. Three individual networks are used for the complete drum transcription. Each network is specified for one of the three main drum instruments: hi-hat, snare drum and bass drum. The three outputs are then combined to provide the final transcription.

3.2 JAR1

The JAR1 network is similar to JAR5, but it has three outputs instead of one: one for each instrument. A single network can then predict the complete transcription for all instruments. The advantage is that the features computed by the first and the second layers are used by all three predictions, resulting in a learning focused on features useful for all instruments.

The layers 1 and 2 compute the features and are used by the three outputs, they have respectively 30 and 70 filters. The layers 3 and 4, which are dense layers, compute the prediction from the features. Thus, the network has 3 independent third layers (of size 256), L_3^{bd} L_3^{sd} and L_3^{hh}, and their corresponding fourth layers, L_4^{bd} L_4^{sd} and L_4^{hh}. Each fourth layer predicts one of the instruments.

3.3 JAR2

The IRC3 network is similar to JAR1, but the learning process is different: if we consider the function the learning process minimizes, $E(\text{labels}, \text{predictions})$, it can actually be decomposed in the sum of three independent functions:

$$\begin{aligned} E(x, y) &= E_{bd} + E_{sd} + E_{hh} \\ &= E(x_{bd}, y_{bd}) + E(x_{sd}, y_{sd}) + E(x_{hh}, y_{hh}) \end{aligned}$$

Each of these function correspond to the output of the network for one of the instruments. In JAR2, an iteration minimizes one of the three function E_{bd}, E_{sd} or E_{hh}, and the next iteration will minimize an other.

Minimizing E_{bd}, for example, will change the weights of layers 1 and 2, since they influence all three outputs, and only the weights of layers L_3^{bd} and L_4^{bd}. The weights of the layers L_3^{sd}, L_3^{hh}, L_4^{sd} and L_4^{hh} are not changed when we minimize E_{bd}.

This allows more specific learning batches, with more positive onsets for the current instrument. To prevent the precision to drop (while the recall increases), the balance between positive and negative onsets in the batch is set to $1/3$ (instead of $1/2$).

3.4 JAR3

Motivated by the fact that JAR5 and JAR2 have shown rather different recall and precision performance on our test set we constructed JAR3 as a combination (bagging) of these two models. The outputs of this system is the mean between the outputs of the three individual network and the corresponding outputs of JAR3.

4. RESULTS

The table 1 displays the F-measure for the indvidual MIREX evaluation subsets and overall (3 classes) results. For the

	JAR5	JAR1	JAR2	JAR3
IDMT	0.62	0.63	0.63	0.64
KT	0.62	0.63	0.63	0.64
RBMA	0.67	0.68	0.70	0.69
MDB	0.66	0.68	0.65	0.67
GEN	0.76	0.81	0.79	0.78
Overall	0.67	0.69	0.68	0.69

Table 1. The averaged F-measure on the evaluation set.

overall, JAR3 and JAR1 showed the best mean f-measure performance with an other algorithm. JAR5 and JAR2 got the fourth and fifth place respectively. The full results can be consulted on MIREX18 website [1].

5. REFERENCES

[1] C. Jacques, A. Roebel: "Automatic drum transcription with convolutional neural networks," *Proceedings of the Digital Audio Effects (DAFx-2018).*

[2] J. Schlüter, S. Böck: "Improved musical onset detection with convolutional neural networks," *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014.

[1] `http://www.music-ir.org/mirex/wiki/2018:Drum_Transcription_Results`