INTERACTIVE ONSET DETECTION IN AUDIO RECORDINGS

Jose J. Valero-Mas, José M. Iñesta
Pattern Recognition and Artificial Intelligence Group
Software and Computing Systems Department
University of Alicante
{jjvalero,inesta}@dlsi.ua.es

ABSTRACT

Onset detection still has room for improvement. State-of-the-art onset detection algorithms achieve good results for a range of applications, but for some situations in which the accuracy is a must, human intervention is required to correct the mistakes committed. In such scheme, accuracy in the result is guaranteed at the expense of the manual correction of all errors. Hence, the issue now lies on finding schemes for efficiently exploiting and reducing that user effort. In this work we present an Interactive Pattern Recognition approach for tackling this issue: using a pre-trained classification-based onset detection algorithm, every time the user corrects an error in the estimation, the system modifies its performance accordingly and recalculates the output. Initial results show that user effort is effectively reduced under our proposal.

1. INTRODUCTION

Given the relevance of onset detection in the Music Information Retrieval (MIR) community, a large amount of algorithms have been proposed to solve this issue. Most detection algorithms base their performance in measuring the change in one or more audio features as, for instance, energy, pitch, phase, or even combinations of all of them. In this sense, very sophisticated approaches have been proposed and, although state-of-the-art algorithms achieve remarkably good results, with these figures it is not possible to claim that the problem has been solved yet.

For situations in which accuracy of onset information is a must, user intervention is eventually required as a post-processing step to manually validate and correct all estimations done by an automatic system. Given that this laborious step cannot be avoided, it seems interesting to consider the use of interactive correction schemes in which the system itself is capable of *learning* from the user corrections and *adapting* its performance for eventually reducing the workload required.

A first example of interactive scheme addressing onset detection and correction is the one presented in [1]. That system, although qualitatively reducing correction workload, is not truly *learning* from the user as it basically changes the estimation parameters of a given onset detection algorithm when errors are pointed out.

In this work we present an interactive onset correction scheme based on Pattern Recognition. By using a *data-driven* scheme we can actually adapt the behaviour of the system by including and/or removing elements from the training set. This update should, in principle, occur every time the user interacts with the system for correcting an error and, once adapted, the model should predict the onset information for time instants after that correction point (previous information is implicitly validated). In this sense, the key point to research is the definition and assessment of different model updating policies.

2. SCHEME PROPOSED

As a starting point we use the onset detection algorithm proposed in [2]. This work considers onset detection as a classification problem: each time frame is considered an instance of a two-class classification problem, *onset* or *non-onset*. Each instance is characterised by 12 descriptors and the analysis parameters are a window size of 4096 samples with a 50 % of overlapping factor. New elements are classified using the k-Nearest Neighbour rule (kNN).

Given that the kNN rule is an instance-based classifier (does not build a model out of the training data), adapting its behaviour is basically modifying its training set, without any need for re-training the model. With this idea in mind, we propose the scheme in Figure 1: the system in [2] retrieves a list of onsets $(\hat{o}_i)_{i=1}^{L}$; as the user corrects the errors, information is added to the *Training Set* of the *Onset Detector*, which modifies its performance; then the detection algorithm recalculates the output; after a number of iterations, the correct list of onsets $(o_i)_{i=1}^{N}$ is retrieved.

The key point in this scheme and in the interactive principle is the sequential sense of the output: given the time dependency of the detection, when the user points an error at position t_{int}, all information located at time frames $t < t_{int}$ is implicitly validated and corrections are therefore only required in time frames $t > t_{int}$. This fact is very important as the user is not only pointing out an error committed by the algorithm (thus clearly stating the need

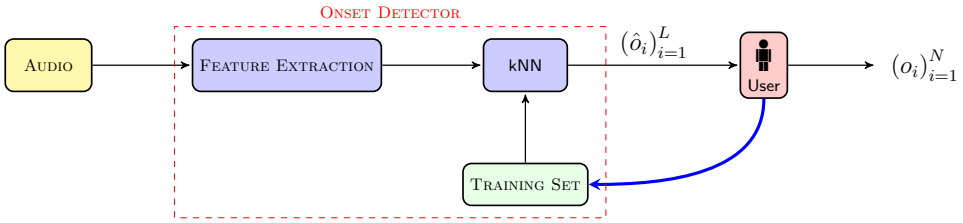

Figure 1. Interactive scheme proposed.

for *learning* that particular case), but also is stating that all previous frames were correctly classified. In this sense, while it seems unarguable the need for including the error point with the correct label as part of the *Training Set*, the question arises with the rest of the information: should the rest of the instances update the *Training Set*? Only some of them? If only some of them are relevant, which ones?

2.1 Interactive policies

To answer those questions we have studied some policies as proof of concept. Assume $\mathbf{x}$ represents the vector of instances of the audio file being analysed. Be t_i a user interaction at frame $\mathbf{x}[t = t_i]$ and let us assume there was a previous interaction t_{i-1} at frame $\mathbf{x}[t = t_{i-1}]$. $\mathbf{M} = [\mathbf{x}[t_{i-1} + 1], ..., \mathbf{x}[t_i - 1]]$ represents the set of instances between the two interactions (the number of vector frames in $\mathbf{M}$ is denoted as $|\mathbf{M}|$).

As aforementioned, the instance $\mathbf{x}[t_i]$ representing the interaction point t_i is always included in the *Training Set* as it is constitutes an error committed by the system and corrected by the user, and thus it should *learn from that*.

Taking that into consideration, we studied four different policies for the elements in $\mathbf{M}$:

1. **Include (INC)**: All elements in $\mathbf{M}$ are included.
2. **Discard (DIS)**: No element in $\mathbf{M}$ is included.
3. **Random selection (RAN)**: $\frac{|\mathbf{M}|}{2}$ elements are randomly selected from $\mathbf{M}$ and included in the set.
4. **Validation (VAL)**: Point $\mathbf{x}[t = t_i]$ is temporary included in the set and $\mathbf{M}$ is used as a *validation set, V*. If the prediction over V when including instance $\mathbf{x}[t = t_i]$ in the *Training Set* is different than previously, point $\mathbf{x}[t = t_i]$ is eventually discarded.

3. INITIAL EXPERIMENTS

For evaluating these policies we have used the PROSEMUS [1] dataset for onset detection using with *leaving-one-out* at file level, i.e. using all files for training except for one used for the interactive evaluation.

User effort has been assessed using Equation 1:

$$R_{GT} = \frac{N_{int}}{N_{GT}} \quad (1)$$

being N_{int} the number of interactions necessary to complete the task and N_{GT} the number of ground-truth onsets.

[1] http://grfia.dlsi.ua.es/cm/projects/prosemus/database.php

Results are shown in Table 1. The case of manually correcting the sequence (MAN) has been added as baseline.

kNN	MAN	INC	RAN	DIS	VAL
1	1.10	**0.74**	0.80	0.86	0.86
5	0.89	**0.66**	0.72	0.80	0.80
9	0.81	**0.65**	0.70	0.76	0.78
11	0.79	**0.65**	0.69	0.75	0.76

Table 1. User effort results. Best scores are highlighted.

Preliminary results show that all interactive policies reduce the amount of necessary interactions (user effort) in the correction with respect to the manual case. The best policy seems to be the one of including all elements in $\mathbf{M}$. Discarding all information in $\mathbf{M}$ increases the necessary effort, suggesting that relevant information is being missed. Random selection of the elements depicts an intermediate result between the two previous policies, which also makes sense as only half of the elements are being included. The last approach does not report any benefit, especially when considering its computational complexity.

A first question that arises is why including all information reports the best results: the system was already able to properly classify those frames but still their inclusion reports a reduction in the effort. A first guess is that class distributions cannot be properly estimated due to the data variability. This idea shall be thoroughly studied and analysed in future work.

Acknowledgements: Work supported by the FPU program of the University of Alicante (UAFPU2014–5883) and the Spanish Ministerio de Economía y Competitividad through project TIMuL (No. TIN2013–48152–C2–1–R, supported by EU FEDER funds).

4. REFERENCES

[1] José M. Iñesta and Carlos Pérez-Sancho. Interactive multimodal music transcription. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, pages 211–215, Vancouver, BC, Canada, 2013.

[2] Jose J. Valero-Mas and José M. Iñesta and Carlos Pérez-Sancho. Onset detection with the user in the learning loop. In *Proceedings of the 7th International Workshop on Music and Machine Learning (MML2014)*, Barcelona, Spain, 2014.