Classification-based Note Tracking for Automatic Music Transcription

Jose J. Valero-Mas[1], Emmanouil Benetos[2], and José M. Iñesta[1]

[1] Pattern Recognition and Artificial Intelligence Group
University of Alicante
{jjvalero,inesta}@dlsi.ua.es
[2] Centre for Digital Music
Queen Mary University of London
emmanouil.benetos@qmul.ac.uk

Abstract. Note tracking constitutes a key process in Automatic Music Transcription as it derives a note-level transcription from a frame-based pitch activation representation. While this stage is commonly performed using a set of hand-crafted rules, this work presents an approach based on supervised classification which automatically infers these policies. An initial frame-level estimation provides the necessary information for segmenting each pitch band in single instances which are later classified as active or non-active note events. Preliminary results using classic classification strategies on a subset of the MAPS piano dataset report an improvement of up to a 15 % when compared to the baseline considered for both frame-level and note-level assessment.

1 Introduction

Automatic Music Transcription (AMT) aims at retrieving a symbolic representation of the musical content present in an audio signal. For doing so, most AMT systems comprise two stages [3]: an initial *multipitch estimation* (MPE) stage in which the system estimates the active pitches in each frame of the signal (frame-level transcription); and a *note tracking* (NT) phase in which the results of the MPE stage are refined to obtain higher-level description of the events in terms of a discrete pitch value, onset and offset (note-level transcription).

While MPE has been largely studied over the years, not so much attention has been paid to the note tracking phase [4]. Thus, note-level transcriptions are most commonly obtained from the frame-level ones by considering combinations of minimum-length pruning filters that remove spurious detections, and gap-filling processes which palliate over-segmentation issues in the pitch activations. Examples in the literature range from rule-based systems [1] to more advanced approaches as, for instance, hidden Markov models (HMM) [7], being onset information occasionally considered to further refine timing issues [6].

This work explores the use of supervised classification to automatically estimate the proper pruning and gap-filling policies for retrieving a note-level transcription. A precedent to this idea may be found in [8] in which a Support Vector

Machine (SVM) classifier is trained on features derived from a Non-Negative Matrix Factorisation (NMF) analysis to perform note tracking. In contrast, the idea in our case is to derive a set of features from both the initial multipitch estimation analysis of an audio piece together with its frame-level transcription obtained from a heuristic approach (cf. to Section 3 for the process considered) and train a binary classifier to further process the events in the frame-level representation as either active or non-active notes. Preliminary results on piano pieces report up to a 15 % of improvement on both frame-level and note-level transcriptions when compared to figures obtained with hand-crafted policies.

2 Proposed method

The method requires the pitch-time posteriorgram $P(p,t)$, corresponding p and t to pitch and time indexes respectively, obtained from a multipitch analysis of an audio piece, an initial frame-level transcription $T_F(p,t)$ and an L-length list of onset events $(o_n)_{n=1}^{L}$. Additionally, let $T_R(p,t)$ be the ground-truth piano-roll representation of the pitch-time activations of the piece.

The binary frame-level transcription $T_F(p,t)$ can be considered a set of $|\mathcal{P}|$ binary sequences of $|t|$ symbols, where $|\mathcal{P}|$ and $|t|$ stand for the total number of pitches and frames in the sequence respectively. In that sense, we may use the elements $(o_n)_{n=1}^{L}$ as delimiters for segmenting each sequence $p_i \in \mathcal{P}$ in $L+1$ subsequences, resulting in a frame-level abstraction quantised by the onsets:

$$T_F(p_i,t) = T_F(p_i, 0 : o_1) \,||\, T_F(p_i, o_1 : o_2) \,||\, ... \,||\, T_F(p_i, o_L : |t| - 1)$$

where $||$ represents the concatenation operator.

Each of these onset-based $L+1$ subsequences per pitch are further segmented to create the instances for the classifier. The delimiters for these segments are the points in which there is a change in the state of the binary sequence, from 0 to 1 or from 1 to 0. Mathematically, for the onset-based subsequence $T_F(p_i, o_n : o_{n+1})$ the $|C|$ state changes are obtained as $C = \{t_m : T_F(p_i,t_m) \neq T_F(p_i,t_{m+1})\}_{t_m=o_n}^{o_{n+1}}$. Thus, the resulting $|C|+1$ segments (instances for the classifier) are:

$$T_F(p_i, o_n : o_{n+1}) = T_F(p_i, o_n : C_1) \,||\, ... \,||\, T_F(p_i, C_{|C|} : o_{n+1}) \;.$$

Figure 1 illustrates this procedure.

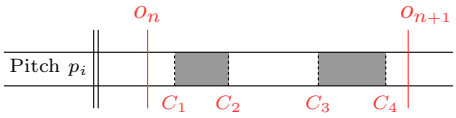

Fig. 1: Segmentation of the onset-based subsequence $T_F(p_i, o_n : o_{n+1})$ into instances. White and grey colours depict sequences of 0 and 1, respectively.

This segmentation process is sufficient for the test corpus, but an additional step is applied for the training one. Instead of exclusively relying on the $T_F(p,t)$ representation, $T_R(p,t)$ is also considered for obtaining the C set by $C = C_{T_F} \cup \{t_m : T_R(p_i,t_m) \neq T_R(p_i,t_{m+1})\}_{t_m=o_n}^{o_{n+1}}$, where C_{T_F} represents the points obtained from $T_F(p,t)$, so that segmentation information includes both ground-truth and estimated data. Labels for the train set are retrieved from $T_R(p,t)$ while labels for instances of the test set are not required since evaluation is eventually done in terms of note tracking (not as classification accuracy).

A set of features is derived out of the segments obtained. It comprises features directly derived from the *geometry* of the instance (i.e., absolute duration or duration relative to the inter-onset interval), others derived from transcription $T_F(p,t)$, as its distance to previous and posterior onsets, and others related to posteriorgram $P(p,t)$ as the average energy in current and octave-related bands. No pitch information is included as feature, thus classification is performed independently of the pitch at issue. Table 1 summarises the features considered.

Table 1: Summary of the features considered. Operator $\langle\cdot\rangle$ retrieves the average value of the elements considered.

Feature	**Definition**	**Description**
Δt	$C_{m+1} - C_m$	Duration of the block
Δo_n	$C_m - o_n$	Distance between previous onset and the starting point of the block
Δo_{n+1}	$o_{n+1} - C_{m+1}$	Distance between end of the block and the posterior onset
$\mathcal{D}$	$\frac{\Delta t}{o_{n+1}-o_n}$	Occupation ratio of the block in the inter-onset interval
E	$\langle P(p_i, C_m : C_{m+1})\rangle$	Mean energy of the multipitch estimation in current band
E_l	$\langle P(p_i - 12, C_m : C_{m+1})\rangle$	Mean energy of the multipitch estimation in previous octave
E_h	$\langle P(p_i + 12, C_m : C_{m+1})\rangle$	Mean energy of the multipitch estimation in next octave

In addition, to incorporate temporal knowledge to the classifier, descriptors of previous and/or posterior instances to the one at issue are considered.

3 Experimentation

For multipitch estimation we consider the Probabilistic Latent Component Analysis (PLCA) system [2], configured to retrieve $|\mathcal{P}| = 88$ pitch values with a temporal resolution of 10 ms. The retrieved pitch-time posteriorgram $P(p,t)$ is processed to obtain a frame-level transcription $T_F(p,t)$ as follows: $P(p,t)$ is normalised to its global maximum so that $P(p,t) \in [0,1]$; then, a median filter

of 70 ms of duration is applied over time to smooth it; after that, it is binarised using a global threshold value of $\theta = 0.1$; finally, a minimum-length pruning filter of 50 ms is applied to remove spurious detections.

In terms of data, the ENSTDkCl set of the MAPS database [5] has been utilised. It consists of 30 pieces played with a Disklavier with their corresponding MIDI files aligned, from which we selected 10. As done in other AMT works, we have only considered the first 30 seconds of each music piece. Additionally, a 5-fold cross validation at the file level has been considered for the experiments.

Regarding onset events, ground-truth information has been extracted from the MIDI files to avoid the influence of the performance of the onset detectors.

As for the classifiers, we have employed four classic strategies as a proof of concept: Decision Trees (DTree), Support Vector Machines (SVM) with first-order polynomial kernel, Nearest Neighbour (1-NN) with Euclidean distance and Decision Tables (DTab) [9]. In reference to the features, we have considered all features in Table 1 for each instance. Experiments have been carried out including features from surrounding instances to test their influence in the system.

As figures of merit, we considered the same metrics as in the MIREX contest: F-measure for both frame-based and onset-based note tracking.

Table 2 shows the results obtained with the classifiers and configurations studied. Baseline considered is the frame-level transcription $T_F(t, p)$.

Table 2: Average and deviation F-measure of the 5-fold cross validation. Notation (x, y) stands for the number of previous and posterior instances considered. Highlighted figures improve the results of the baseline configuration.

Scheme	$(0,0)$		$(1,1)$		$(2,2)$	
	Frame	**Note**	**Frame**	**Note**	**Frame**	**Note**
Baseline	0.51 ± 0.06	0.52 ± 0.06	0.51 ± 0.06	0.52 ± 0.06	0.51 ± 0.06	0.52 ± 0.06
DTree	$\mathbf{0.66 \pm 0.05}$	$\mathbf{0.64 \pm 0.04}$	$\mathbf{0.64 \pm 0.04}$	$\mathbf{0.60 \pm 0.05}$	$\mathbf{0.64 \pm 0.05}$	$\mathbf{0.60 \pm 0.06}$
SVM	0.50 ± 0.06	$\mathbf{0.60 \pm 0.06}$	0.51 ± 0.08	$\mathbf{0.63 \pm 0.05}$	$\mathbf{0.53 \pm 0.09}$	$\mathbf{0.64 \pm 0.06}$
1-NN	$\mathbf{0.54 \pm 0.09}$	0.52 ± 0.05	$\mathbf{0.53 \pm 0.08}$	0.52 ± 0.09	0.51 ± 0.05	0.51 ± 0.04
DTab	$\mathbf{0.60 \pm 0.07}$	$\mathbf{0.60 \pm 0.02}$	$\mathbf{0.56 \pm 0.07}$	$\mathbf{0.57 \pm 0.04}$	$\mathbf{0.56 \pm 0.07}$	$\mathbf{0.57 \pm 0.03}$

Results in Table 2 show that the Decision Tree and Decision Table classifiers consistently outperform baseline figures for both metrics. SVM improves the note-level metric, while frame-level results rarely differ from baseline, which may be due to the basic kernel considered. 1-NN does not achieve remarkable results as figures obtained consistently tie with baseline, probably due to a lack of model generalisation. Overall, it is important to remark that the improvement shown by most classifiers for the note-level metric justifies the exploration of this approach.

Regarding the use of additional features from adjacent instances, no strong conclusions can be derived: while in the case of Decision Trees note-level results suffer a drop when adding features from adjacent instances, SVM shows the opposite trend since the same metric improves as more features are added.

4 Conclusions and future work

This work explored the use of a classification paradigm to perform note tracking by combining features derived from a multipitch analysis, an initial frame-level transcription and onset information. The experimentation carried out showed promising results as the use of the proposed approach remarkably improved the results on up to a 15 % when compared to the baseline considered.

Future work considers the study of training set optimisation techniques such as prototype selection to improve the generalisation of the instance-based classifiers and the use of time-aware classifiers such as Recurrent Neural Networks. Additional conclusions may be drawn from considering estimated onset events and other timbres to assess the generalisation capabilities of the proposal.

Acknowledgements

Work supported by Universidad de Alicante through FPU program (UAFPU2014--5883) and the Spanish Ministerio de Economía y Competitividad through project TIMuL (No. TIN2013–48152–C2–1–R, supported by EU FEDER funds). EB is supported by a UK RAEng Research Fellowship (grant no. RF/128).

References

1. Bello, J.P., Daudet, L., Sandler, M.B.: Automatic Piano Transcription Using Frequency and Time-Domain Information. IEEE Trans. Audio Speech Lang. Process. 14(6), 2242–2251 (2006)
2. Benetos, E., Cherla, S., Weyde, T.: An efficient shift-invariant model for polyphonic music transcription. In: 6th International Workshop on Machine Learning and Music (MML). Prague, Czech Republic (September 2013)
3. Benetos, E., Dixon, S., Giannoulis, D., Kirchhoff, H., Klapuri, A.: Automatic music transcription: challenges and future directions. J. Intell. Inf. Syst. 41(3), 407–434 (2013)
4. Duan, Z., Temperley, D.: Note-level Music Transcription by Maximum Likelihood Sampling. In: Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR). pp. 181–186. Taipei, Taiwan (October 2014)
5. Emiya, V., Badeau, R., David, B.: Multipitch Estimation of Piano Sounds Using a New Probabilistic Spectral Smoothness Principle. IEEE Trans. Audio Speech Lang. Process. 18(6), 1643–1654 (2010)
6. Iñesta, J.M., Pérez-Sancho, C.: Interactive multimodal music transcription. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 211–215. Vancouver, Canada (May 2013)
7. Poliner, G.E., Ellis, D.P.: A Discriminative Model for Polyphonic Piano Transcription. EURASIP J. Adv. Signal Process. 2007(1) (2007)
8. Weninger, F., Kirst, C., Schuller, B., Bungartz, H.: A discriminative approach to polyphonic piano note transcription using supervised non-negative matrix factorization. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 6–10. Vancouver, Canada (May 2013)
9. Witten, I.H., Frank, E., Hall, M.A.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann Publishers Inc., USA, 3rd edn. (2011)