PAPER TEMPLATE FOR ISMIR 2014

First author
Affiliation1
author1@ismir.edu

Second author
Retain these fake authors in submission to preserve the formatting

Third author
Affiliation3
author3@ismir.edu

ABSTRACT

The abstract should be placed at the top left column and should contain about 150-200 words.

1. INTRODUCTION

This template includes all the information about formatting manuscripts for the ISMIR 2014. Please follow these guidelines to give the final proceedings a uniform look. If you have any questions, please contact the Conference Management. This template can be downloaded from the ISMIR 2014 web site (http://ismir2014.ismir.net).

2. PAGE SIZE

The proceedings will be printed on portrait A4-size paper (21.0cm x 29.7cm). All material on each page should fit within a rectangle of 17.2cm x 25.2cm, centered on the page, beginning 2.0cm from the top of the page and ending with 2.5cm from the bottom. The left and right margins should be 1.9cm. The text should be in two 8.2cm columns with a 0.8cm gutter. All text must be in a two-column format. Text must be fully justified.

3. TYPESET TEXT

3.1 Normal or Body Text

Please use a 10pt (point) Times font. Sans-serif or non-proportional fonts can be used only for special purposes, such as distinguishing source code text.

The first paragraph in each section should not be indented, but all other paragraphs should be.

3.2 Title and Authors

The title is 14pt Times, bold, caps, upper case, centered. Authors' names are omitted when submitting for double-blind reviewing. The following is for making a camera-ready version. Authors' names are centered. The lead author's name is to be listed first (left-most), and the co-authors' names after. If the addresses for all authors are the same, include the address only once, centered. If the authors have different addresses, put the addresses, evenly spaced, under each authors' name.

3.3 First Page Copyright Notice

Please include the copyright notice exactly as it appears here in the lower left-hand corner of the page. It is set in 8pt Times.

3.4 Page Numbering, Headers and Footers

Do not include headers, footers or page numbers in your submission. These will be added when the publications are assembled.

4. FIRST LEVEL HEADINGS

First level headings are in Times 10pt bold, centered with 1 line of space above the section head, and 1/2 space below it. For a section header immediately followed by a subsection header, the space should be merged.

4.1 Chord Recognition

Despite being one of the oldest MIREX tasks, evaluation methodology and metrics for automatic chord recognition is an ongoing topic of discussion. Several recent articles address issues and concerns with vocabularies, comparison semantics, and other lexicographical challenges unique to chord recognition []. Ultimately, the source of this difficulty stems from the inherent subjectivity in "spelling" a chord name and the level of detail a human observer can provide in a reference annotation [**?**]. As a result, a consensus has yet to be reached regarding the single best approach to comparing two sequences of chord labels, and instead are often compared over a set of rules, e.g Major-Minor, Sevenths, with or without inversions, and so on.

Thanks to the previous efforts of Harte [], text representations of chord labels adhere to a standardized format, consisting of a root, quality, extensions, and a bass note; of these, only the root is strictly required. However, in order to efficiently compare chords in a variety of different ways, it is helpful to first translate a given chord label $\mathcal{C}$ into a numerical representation, shown in Figure **??**. In this example, a $G:7(9)/5$ is mapped to split into 4 pieces of information: one, the root is mapped to an absolute pitch class $\mathcal{R}$, in $[0, 11]$, where $C \rightarrow 0$, $C\sharp/D\flat \rightarrow 1$, etc; two, the quality is mapped to a root-invariant 12-dimensional bit vector $\mathcal{Q}$ by setting the scale degrees of the quality; three, any extensions are applied (via addition or omission) to the quality bit vector as scale degrees in a single octave, resulting in pitch vector $\mathcal{P}$; and four, the bass interval (5) is translated to the relative scale degree in semitones $\mathcal{B}$. Note that the add-9 is rolled into a single octave as an add-2. This is a matter of convenience as extended chords (9's, 11's or 13's) are traditionally resolved to a single-octave equivalent, but the bit-vector representation could be easily expanded to represent such information.

Having gone through this bit of effort, it is now straightforward to compare chords along the five rules used in MIREX 2013:

1. Root:
   (a) $\mathcal{R}_{est} == \mathcal{R}_{ref}$
   (b) $\forall \mathcal{Q}_{ref}$
2. Major-Minor: Rule 1.a, plus
   (a) $\mathcal{Q}_{est} == \mathcal{Q}_{ref}$
   (b) $\mathcal{Q}_{ref} \in \{Maj, min\}$
3. Major-Minor w/Inversions: Rule 2, plus
   (a) $\mathcal{B}_{ref} \in \mathcal{Q}_{ref}$
4. Sevenths: Rule 1.1, plus
   (a) $\mathcal{Q}_{est} == \mathcal{Q}_{ref}$
   (b) $\mathcal{Q}_{ref} \in \{Maj, min, Maj7, min7, 7\}$
5. Sevenths w/Inversions: Rule 4, plus
   (a) $\mathcal{B}_{ref} \in \mathcal{Q}_{ref}$

Following recent trends in MIREX, an overall score is computed by weighting each comparison by the duration of its interval, over all intervals; stated another way, this is the piecewise continuous-time integral of the intersection of two chord sequences, $(\mathbf{C}_{ref}, \mathbf{C}_{est})$, expressed as follows:

$$S(\mathbf{C}_{ref}, \mathbf{C}_{est}) = \frac{1}{T} \int_{t=0}^{T} \mathcal{C}_{ref}(t) == \mathcal{C}_{est}(t) \quad (1)$$

Here, this is achieved here by forming the union of the boundaries in each sequence, and summing the time intervals of the correct ranges. Note that equivalence is subject to one of the rules defined previously.

Finally, the total score over a set of N items is given by a discrete summation, where the importance of each score, S_n, is weighted by the duration, T_n, of each annotation:

$$S_{total} = \frac{\sum_{n=0}^{N} T_n * S_n}{\sum_{n=0}^{N} T_n} \quad (2)$$

4.2 Segmentation

Evaluation criteria for segmentation fall into two categories: boundary annotation, and structural annotation. *Boundary annotation* is the task of predicting the times at which structural changes occur, such as when a *verse* transitions to a *refrain*. *Structural annotation* is the task of assigning labels to detected segments. The estimated labels may be arbitrary strings — such as A, B, C, *etc.* — and they need not describe functional concepts.

In both tasks, we assume that annotations take the form of a partitioning of the track into intervals $(s_i, t_i)_{i=1}^{m}$, where $s_0 = 0$ denotes the beginning of track, t_m denotes the end of the track, and $t_i = s_{i+1}$. Structural annotation additionally requires that each interval be assigned a label y_i.

4.2.1 Boundary annotation

Within boundary annotation, there are two categories of evaluation metrics: detection, and deviation [6].

Boundary detection measures the precision, recall, and f-measure of boundary prediction within a tolerance window. Let B^R (B^E) denote the set of unique interval boundaries in the reference (estimated) annotation, and let W denote a tolerance window, typically either 0.5s or 3.0s. A *hit* is defined as a pair $b_e \in B^E$, $b_r \in B^R$ such that $|b_e - b_r| \leq W$.

No estimated boundary is counted as a hit for more than one reference boundary, and vice versa. This is accomplished by computing a maximum bipartite matching H_W between B^E and B^R (subject to the window constraint W) using the Hopcroft-Karp algorithm [1]. Precision and recall with window W are defined as

$$P_W(B^R, B^E) := \frac{|H_W|}{|B^E|} \quad (3)$$

$$R_W(B^R, B^E) := \frac{|H_W|}{|B^R|}, \quad (4)$$

and the corresponding f-measure is defined by taking their harmonic mean.

Boundary deviation is comprised of two scores, which measure the median time between a reference boundary and its nearest corresponding estimated boundary, and vice versa. These two metrics have previously been defined as *true-to-predicted* (T-to-P) and *predicted-to-true* (P-to-T), though in the terminology of this document, we use alternative naming of *reference-to-estimated* (R-to-E) and *estimated-to-reference* (E-to-R):

$$\text{R-to-E}(B^R, B^E) := \operatorname*{median}_{b_r \in B^R} \min_{b_e \in B^E} |b_r - b_e| \quad (5)$$

$$\text{E-to-R}(B^R, B^E) := \operatorname*{median}_{b_e \in B^E} \min_{b_r \in B^R} |b_r - b_e| \,. \quad (6)$$

4.2.2 Structural annotation

Two standard methods of evaluating structural annotation accuracy are *pairwise classification* [2] and conditional entropy [3]. In both methods, a collection of samples are generated by sampling the labels at time steps between 0 and the track duration. Each sample x_i is assigned a reference label y_i^R and and estimated label y_i^E. Our implementation follows the MIREX guidelines, and generates samples at a default rate of 10Hz.

In order for annotation metrics to be well-defined, both reference and estimated annotations must span the same duration of time. This is accomplished in our implementation by trimming or padding the estimated annotation to exactly match the start and end-times reported in the reference annotation, and synthesizing unique reference labels if necessary. Note that the reference annotation is never modified, only the estimated annotations: in this way, comparison of multiple estimations cannot be compromised by an estimator reporting an incorrect track duration.

Given the labels for the samples $\{(y_i^R, y_i^E)\}_{i=1}^n$, the *pairwise classification* metrics are defined as precision, recall, and f-measure of label agreement over all unique, distinct pairs $i \neq j$. Let $A_R = \{\{i,j\} \mid y_i^R = y_j^R\}$ denote the set of similarly labeled frames in the reference, with A_E defined analogously for the estimation. Then the precision and recall are defined simply as

$$P_{\text{pair}} := \frac{|A_E \cap A_R|}{|A_E|} \tag{7}$$

$$R_{\text{pair}} := \frac{|A_E \cap A_R|}{|A_R|}. \tag{8}$$

Normalized conditional entropy scores are computed by estimating the conditional entropy of the estimated label y^E given the reference label y^R and vice versa. Let P denote the empirical joint distribution over reference and estimated labels:

$$P_{ij} \propto \left|\{t \mid y_t^R = i \wedge y_t^E = j\}\right|. \tag{9}$$

Then the conditional entropies $H(E \mid R)$ and $H(R \mid E)$ (in bits) are derived from the marginals $p_i^R = \sum_j P_{ij}$ and $p_j^E = \sum_i P_{ij}$ as described by Lukashevich [3].

The final scores are defined as *over-segmentation* (S_{O}) and *under-segmentation* (S_{U}):

$$S_{\text{O}} := 1 - \frac{H(E \mid R)}{\log_2 |\ell^E|} \tag{10}$$

$$S_{\text{U}} := 1 - \frac{H(R \mid E)}{\log_2 |\ell^R|}, \tag{11}$$

where ℓ^R and ℓ^E denote the sets of unique label values given in the reference and estimation.

4.3 Melody Extraction

4.3.1 Task definition

Melody extraction algorithms aim to produce a sequence of frequency values corresponding to the pitch of the dominant melody from a musical recording [5]. The estimated pitch is represented as a time series of fundamental frequency (f_0) values in Hz sampled on a fixed time grid (e.g. every 10 ms). To evaluate the estimated sequence, a reference sequence is generated by running a pitch tracker on the monophonic melody track (requiring access to the multi-track recording session) and manually correcting any mistakes made by the pitch tracker. The estimate is then evaluated against the reference by comparing the two frequency sequences on a frame-by-frame basis, and computing five global measures, first used in the MIREX 2005 AME evaluation [4]. The goal of these measures, defined below, is to assess the algorithm's performance on two subtasks of melody extraction: (1) pitch estimation, i.e. how well the algorithm estimates the pitch of the melody, and (2) voicing detection, i.e. how well the algorithm determines when the melody is present in a frame (a *voiced* frame) or absent (an *unvoiced* frame). To allow evaluation of these two subtasks independently, a melody extraction algorithm can provide a frequency estimate even for frames it has determined to be unvoiced.

4.3.2 Evaluation measures

The following definitions are taken from [5] with permission from the authors. Let the estimated melody pitch frequency vector be $\mathbf{f}$, and the true sequence (reference) be $\mathbf{f}^*$. Let us also define a voicing indicator vector $\mathbf{v}$, whose τ^{th} element $v_\tau = 1$ when a melody pitch is detected, with corresponding ground truth $\mathbf{v}^*$. We also define an "unvoicing" indicator $\bar{v}_\tau = 1 - v_\tau$. Recall that an algorithm may report an estimated melody pitch ($f_\tau > 0$) even for times where it reports no voicing ($v_\tau = 0$). Then the measures are:

- **Voicing Recall Rate**: The proportion of frames labeled as melody frames in the reference that are estimated as melody frames by the algorithm.

$$\text{Rec}_{\text{vx}} = \frac{\sum_\tau v_\tau v_\tau^*}{\sum_\tau v_\tau^*} \tag{12}$$

- **Voicing False Alarm Rate**: The proportion of frames labeled as non-melody in the reference that are mistakenly estimated as melody frames by the algorithm.

$$\text{FA}_{\text{vx}} = \frac{\sum_\tau v_\tau \bar{v}_\tau^*}{\sum_\tau \bar{v}_\tau^*} \tag{13}$$

- **Raw Pitch Accuracy**: The proportion of melody frames in the reference for which f_τ is considered correct (i.e. within half a semitone of the ground truth f_τ^*).

$$\text{Acc}_{\text{pitch}} = \frac{\sum_\tau v_\tau^* \mathcal{T}\left[\mathcal{M}(f_\tau) - \mathcal{M}(f_\tau^*)\right]}{\sum_\tau v_\tau^*} \tag{14}$$

where $\mathcal{T}$ is a threshold function defined by:

$$\mathcal{T}[a] = \begin{cases} 1 & \text{if } |a| < 0.5 \\ 0 & \text{if } |a| \geq 0.5 \end{cases} \tag{15}$$

and $\mathcal{M}$ maps a frequency in Hertz to a melodic axis as a real-valued number of semitones above an arbitrary reference frequency f_{ref}:

$$\mathcal{M}(f) = 12 \log_2 \left(\frac{f}{f_{\text{ref}}}\right) \tag{16}$$

- **Raw Chroma Accuracy**: As raw pitch accuracy, except that both the estimated and reference f_0 sequences are mapped onto a single octave. This gives a measure of pitch accuracy which ignores octave errors, a common error made by melody extraction systems:

$$\text{Acc}_{\text{chroma}} = \frac{\sum_\tau v^*_\tau \mathcal{T}\left[\langle \mathcal{M}(f_\tau) - \mathcal{M}(f^*_\tau)\rangle_{12}\right]}{\sum_\tau v^*_\tau} \tag{17}$$

Octave equivalence is achieved by taking the difference between the semitone-scale pitch values modulo 12 (one octave), where

$$\langle a \rangle_{12} = a - 12\lfloor \frac{a}{12} + 0.5 \rfloor. \tag{18}$$

- **Overall Accuracy**: this measure combines the performance of the pitch estimation and voicing detection tasks to give an overall performance score for the system. It is defined as the proportion of all frames correctly estimated by the algorithm, where for non-melody frames this means the algorithm labeled them as non-melody, and for melody frames the algorithm both labeled them as melody frames and provided a correct f_0 estimate for the melody (i.e. within half a semitone of the reference):

$$\text{Acc}_{\text{ov}} = \frac{1}{L}\sum_\tau v^*_\tau \mathcal{T}\left[\mathcal{M}(f_\tau) - \mathcal{M}(f^*_\tau)\right] + \bar{v}^*_\tau \bar{v}_\tau \tag{19}$$

where L is the total number of frames.

The performance of an algorithm on an entire music collection for a given measure is obtained by averaging the per-excerpt scores for that measure over all excerpts in the collection.

4.3.3 Discussion

In the measure definitions provided above, it is assumed that both the estimate and reference sequences are sampled using the same time grid (hop size). In practice, however, this is not always the case, since the time grid of the reference depends on the hop size used by the pitch tracker that produced it, and similarly the time grid of the estimate depends on the specific melody extraction algorithm that produced it. This means that the sequences must be resampled onto a common time-grid prior to the computation of the measures. For the MIREX AME task, any sequence (be it reference or estimate) that is not sampled using a 10 ms hop size is resampled using 0^{th}-order interpolation, i.e. each frequency value in the target sequence is set to its nearest neighbour (in time) from the source sequence. This kind of interpolation can potentially have a detrimental effect on the evaluation, depending on the difference between the source and target time grids. In particular, it can result in artificially low scores for sequences with rapidly changing pitch values, such as opera singing with deep vibrato.

For this reason, the melody evaluator in mir_eval uses 1^{st}-order (linear) interpolation by default in order to map the reference and estimate sequences onto a common time-grid. Assuming the original timestamps of both sequences correspond to the *center* of each analysis frame (as they should), using 1^{st}-order rather than 0^{th}-order interpolation means the results returned by mir_eval are lower-bounded by the MIREX results and are, in our view, more accurate.

5. REFERENCES

[1] John E Hopcroft and Richard M Karp. An n^5/2 algorithm for maximum matchings in bipartite graphs. *SIAM Journal on computing*, 2(4):225–231, 1973.

[2] Mark Levy and Mark Sandler. Structural segmentation of musical audio by constrained clustering. *Audio, Speech, and Language Processing, IEEE Transactions on*, 16(2):318–326, 2008.

[3] Hanna M Lukashevich. Towards quantitative measures of evaluating song segmentation. In *ISMIR*, pages 375–380, 2008.

[4] G. E. Poliner, D. P. W. Ellis, A. F. Ehmann, E. Gómez, S. Streich, and B. Ong. Melody transcription from music audio: Approaches and evaluation. *IEEE Trans. on Audio, Speech, and Language Processing*, 15(4):1247–1256, 2007.

[5] J. Salamon, E. Gómez, D. P. W. Ellis, and G. Richard. Melody extraction from polyphonic music signals: Approaches, applications and challenges. *IEEE Signal Processing Magazine*, 31(2):118–134, Mar. 2014.

[6] Douglas Turnbull, Gert RG Lanckriet, Elias Pampalk, and Masataka Goto. A supervised approach for detecting boundaries in music using difference features and boosting. In *ISMIR*, pages 51–54, 2007.