DOWNBEAT DETECTION WITH CONDITIONAL RANDOM FIELDS AND DEEP LEARNED FEATURES

Simon Durand, Slim Essid
LTCI, CNRS, Télécom ParisTech
Université Paris-Saclay, 75013, Paris, France
simon.durand@telecom-paristech.fr

ABSTRACT

In this paper, we introduce a novel Conditional Random Field (CRF) system that detects the downbeat sequence of musical audio signals. Feature functions are computed from four deep learned representations based on harmony, rhythm, melody and bass content to take advantage of the high-level and multi-faceted aspect of this task. Downbeats being dynamic, the powerful CRF classification system allows us to combine our features with an adapted temporal model in a fully data-driven fashion. Some meters being under-represented in our training set, we show that data augmentation enables a statistically significant improvement of the results by taking into account class imbalance. An evaluation of different configurations of our system on nine datasets shows its efficiency and potential over a heuristic based approach and four downbeat tracking algorithms.

1. INTRODUCTION

Musical rhythm can often be organized in several hierarchical levels. These levels don't always correspond to musical events and have a regular temporal interval that can change over time to follow the musical tempo. One of these levels is the tatum level and is at the time scale of onsets. The next one is often the beat level and can be intuitively understood as the hand clapping or foot tapping level. Then in several music traditions there is the bar level that groups beats of different accentuation together. The first beat of the bar is called the downbeat. The aim of this work is to automatically find the downbeat positions from musical audio signals. The downbeat is useful to musicians, composers and conductors to segment, navigate and understand music more easily. Its automatic estimation is also useful for various applications in music information retrieval, computer music and computational musicology.

This task is receiving more attention recently. With the increasing number of annotated music files and refined learning strategies, methods using probabilistic models and machine learning algorithms tend to be the most successful [4, 13, 18]. Once a downbeat detection function has been extracted, most systems use a temporal model to take advantage of the structured organization of downbeats and output the downbeat sequence. It includes heuristics [3], dynamic programming [25], hidden Markov models [23] and particle filters [18] among others.

In this work, we propose for the first time a Conditional Random Field (CRF) framework for the task of downbeat tracking. First, four complementary features related to harmony, rhythm, melody and bass content are extracted and the signal is segmented at the tatum level. Adapted convolutional neural networks (CNN) to each feature characteristics are then used for feature learning. Finally, a feature representation concatenated from the last and/or penultimate layer of those networks is used to define observation feature functions and is fed into a Markovian form of CRF that will output the downbeat sequence.

1.1 Related work

A CRF framework is used in [7] and [16] for the field of beat tracking. However, the optimal weights of the observations and transitions feature functions are not directly learned from the data.

The system presented in [14] uses an interesting idea of limiting engineered hypotheses by segmenting the data in onsets and learning the activation of downbeats with a Support Vector Machine classifier. Contrary to our work, it requires manual annotation of either the first part of the tested song or of a very similar song and outputs an intermediary downbeat activation function as opposed to the final downbeat positions.

In [6], the same segmentation, low-level feature extraction and complementary CNNs are used. However, the proposed system includes three main differences:

- We are not only using an individual output per downbeat candidate but a detailed high-level representation also coming from the penultimate layer of the neural networks. Besides, we don't optimize individual features on isolated downbeat occurrences, but features from all the deep networks simultaneously on a whole structured downbeat sequence.

- We are using another type of classifier, namely CRF, known to be more effective than Hidden Markov

Models especially in high dimensional settings due to being a discriminative classifier.

- A fully data driven approach, after extracting low-level features, is adopted. It takes advantage of data augmentation procedures to allow for a proper training of the CRF classifiers, limiting the use of ad-hoc heuristics and hand-crafted data transformations.

2. FEATURE LEARNING

The feature learning part of our system is the same as in [6]. We first segment the audio signal in tatums as seen in figure 1. We then simplify the downbeat detection task to a classification problem where the goal is to find which tatums are at a downbeat position. Human perception of downbeats depending on several musical cues, we then extract four low-level features related to melody, rhythm, harmony and bass content. Each low-level feature input, shown in figure 2, is fed to a convolutional neural network adapted to its characteristics. The bass content neural network (BCNN) targets melodic and percussive bass instruments. The melodic neural network (MCNN) targets relative melodic patterns which are known to play a role in human perception of meter regardless of their absolute pitch [29] with max pooling. The harmonic neural network (HCNN) learns how to detect harmonic change in the input and is trained on all different harmony transposition by data augmentation. Finally, the rhythmic neural network (RCNN) aims at learning length specific rhythmic patterns with multi-label learning, instead of sudden changes in the rhythm feature that are not very indicative of a downbeat position. For more details about the motivations behind the design choices made for each network the interested reader is referred to [6].

3. CRF SYSTEM FOR DOWNBEAT TRACKING

Two high-level feature representations coming from the last and penultimate layer of each network are then used as input to a Conditional Random Field (CRF) classifier.

3.1 CRF-based classification

CRF [19] are a powerful class of discriminative classifiers for structured input–structured output data prediction, which have proven successful in a variety of real-world classification tasks [26, 27] and also in combination with neural networks [24]. They model directly the posterior probabilities of output sequences $\underline{\mathbf{y}} = (y_1, \cdots, y_n)$ given input observation sequences $\underline{\mathbf{x}} = (\mathbf{x}_1, \cdots, \mathbf{x}_n)$ according to:

$$p(\underline{\mathbf{y}}|\underline{\mathbf{x}};\boldsymbol{\theta}) = \frac{1}{Z(\underline{\mathbf{x}},\boldsymbol{\theta})} \exp \sum_{j=1}^{D} \theta_j G_j(\underline{\mathbf{x}},\underline{\mathbf{y}})$$

where $G_j(\underline{\mathbf{x}},\underline{\mathbf{y}})$ are feature functions describing the observations, θ_j are the model parameters (assembled as $\boldsymbol{\theta} = [\theta_j]_{1\leq j\leq D}$), and $Z(\underline{\mathbf{x}})$ is a normalizing factor that guarantees that $p(\underline{\mathbf{y}}|\underline{\mathbf{x}})$ is a well defined probability, which sums to 1.

Owing to the sequential nature of the downbeat classification problem, we use a Markovian form of CRF, where the transition feature functions, denoted by t_j, are defined on two consecutive labels, in a linear-chain fashion, and observation feature functions, denoted by v_j, depend on single labels, so that:

$$p(\underline{\mathbf{y}}|\underline{\mathbf{x}};\boldsymbol{\theta}) = \frac{1}{Z(\underline{\mathbf{x}},\boldsymbol{\theta})} \exp \left\{ \sum_{i=1}^{n} \sum_{j=1}^{D_o} \theta_j v_j(y_i, \underline{\mathbf{x}}, i) + \sum_{i=1}^{n} \sum_{j=1}^{D_t} \theta_j t_j(y_{i-1}, y_i, \underline{\mathbf{x}}, i) \right\}. \quad (1)$$

More specifically, the transition feature functions we use are such that $t_j(y_{i-1} = k, y_i = l, \underline{\mathbf{x}}, i) = \mathbb{I}(y_i = l)\mathbb{I}(y_{i-1} = k)$, where $\mathbb{I}(.)$ is the indicator function (equal to 1 if its argument is true and otherwise equal to 0). As for the observation feature functions they are chosen to be of the form $v_j(y_i = l, \underline{\mathbf{x}}, i) = e_j \mathbb{I}(y_i = l)$ where e_j are obtained by the feature representation learned by the networks presented in section 2. Actually, two schemes are envisaged here. In the first variant, the e_j features are taken to be directly the final outputs of the bass, melodic, harmonic and rhythmic networks. Alternatively, we added the output of the penultimate layer [1] which can be viewed as lower level features that were optimized, as part of the network training processes, to discriminate downbeats from tatums. The deep network penultimate layer output is a powerful feature representation that can be used as an input to a dedicated classifier to improve accuracy [9]. The last layer of our networks being essentially a linear combination of the penultimate layer features followed by a normalization to map them to probabilities, the CRF classifier is a good fit for the final weighting of those features, based on the more optimal output-sequence level maximum a posteriori criterion $p(\underline{\mathbf{y}}|\underline{\mathbf{x}};\theta)$, compared to the static criterion optimized in the last layer of the networks. The harmonic network penultimate layer dimension is of 1000 and each of the other networks penultimate layer dimension is of 800.

3.2 Defining the output-space

The set of output labels Y_i^j represents the position i of a tatum in a j tatum-long bar, with $i \in \{1...j\}$ and $j \in \{3, 4, 5, 6, 7, 8, 9, 10, 12, 16\}$. We consider an additional label for bars containing more than 16 tatums for a total of 81 labels. This way, the feature function weights depend on the bar length, in tatums, and the position inside the bar. For instance, the sixth tatum of a 6 tatum-long bar Y_6^6 and the sixth tatum of a 8 tatum-long bar Y_6^8 have different musical properties. In the first case, we want the transition feature functions to emphasize the next output to be the first tatum of a 6 tatum-long bar Y_1^6. In the second case, we want to emphasize the next output to be the seventh tatum of a 8 tatum-long bar instead Y_7^8. The observation features, taking into account one or two bars of

[1] before the ReLU to keep information about negative units

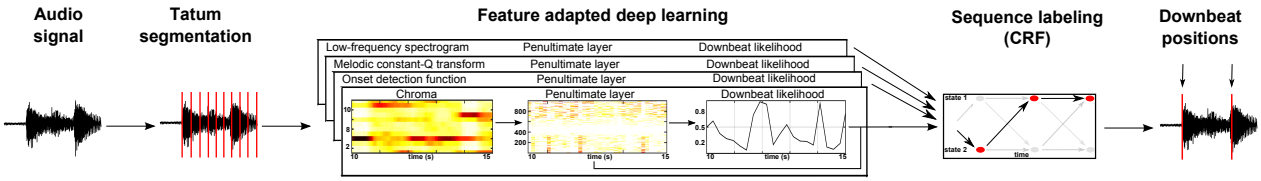

Figure 1. Model overview. The signal is quantized in tatums. Four low-level features related to harmony, rhythm, melody and bass content are extracted. High-level feature representations are learned with four convolutional networks adapted to each feature characteristics. The networks penultimate layer, along with the downbeat likelihood, are fed in a CRF to find the downbeat sequence among the tatums.

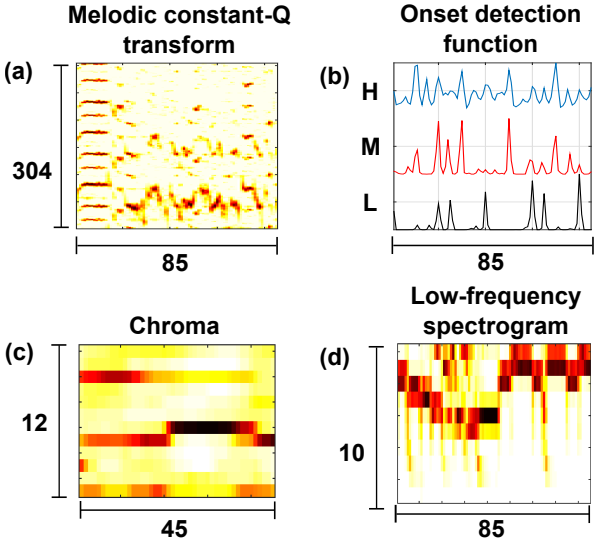

Figure 2. Low-level features with their temporal and spectral dimension, used as input of the melodic neural network (a), rhythmic neural network (b), harmonic neural network (c) and bass content neural network (d).

temporal context, will also be rather different and are better treated separately. It is therefore important to distinguish those two outputs for a consistent decoding.

3.3 Labeling the training data

The training data being only annotated in beat and downbeat, defining its labels is not straightforward. First, bars of 2, 11, 13, 14 and 15 tatums are not present in our model for efficiency, robustness and because they are barely present in most music datasets but they can't be ignored to train the model efficiently. They are then annotated to the most common neighbor bar-length: 14 and 15 tatum-long bars are annotated as 16 tatum-long bars. 11 and 13 tatum-long bars are annotated as 12 tatum-long bars. The last state of those metrical levels is either removed or repeated to do so. 2 tatum-long bars are annotated as 3 tatum-long bars if the following bar is a 3 tatum-long bar for continuity or as 4 tatum-long bar otherwise as they are the most common neighbor for a duple meter.

Second, the beginning and end of songs are sometimes not properly estimated or annotated and considering or ignoring all observations before the first or after the last downbeat can lead to training problems. For the beginning of songs, we removed the samples that where more than one bar before the first annotated downbeat as they were not reliable enough. We then annotated the bar preceding the first downbeat with the same classes than the bar containing the first downbeat for continuity, and finally removed samples in this first bar randomly. It allows the initialization of the position inside the bar to be randomized. The procedure is applied in reverse for the end of songs.

Although extensive tests were not performed, we obtain a gain in performance by about 4 percent points (pp) by using this annotation process compared to a simple representation of all these non conventional cases by an additional label.

3.4 Handling class-imbalance with data augmentation

Not all metrical level are well represented in the used datasets. In fact, {3,4,6,8,12,16} tatum-long bars, i.e. bars of 3 and 4 beats, represent more than 96% of the data and will be the focus of the CRF model. In this subset, bars of 3 beats will be represented by 3, 6, and roughly half of 12 tatum-long bars. This represents approximately 15% of the data. Those metrical levels are then non negligible but under-represented. Such data imbalance is known to create difficulties while training classifiers like CRFs. We therefore balance our dataset with data augmentation. We use time-stretching by a factor of 1.1 and 0.9 and pitch shifting by ± 1 semitone on 3-beats-per-bar songs to do so. The implementation is done thanks to the *muda* package presented in [20]. We will study in the experiments the added value of the data augmentation.

4. EXPERIMENTAL SETUP

4.1 Evaluation methods

We use the F-measure and a statistical test to assess the performance of our system:

F-measure: The F-measure is the harmonic mean of the precision (ratio of detected downbeat that are relevant) and the recall (the ratio of relevant downbeat detected). It is an instantaneous measure of performance that is used in the MIREX downbeat tracking evaluation [2]. We use a

[2] http://www.music-ir.org/mirex/wiki/2016:

tolerance window of $\pm$70ms. The configuration with the best F-measure with be highlighted in bold. We do not take into account the first 5 seconds and last 3 seconds of audio in our evaluation metric since the annotation is sometimes missing or not very reliable there.

Statistical tests: To assess statistical significance, we perform a Friedman's test and a Tukey's honestly significant criterion (HSD) test with a 95% confidence interval. System(s) with a statistically significant improvement over the rest on the whole dataset will be underlined.

4.2 Databases

We use nine different databases in this work, for a total of 1511 audio tracks of about 43 hours of audio music. Using multiple datasets allows us to see the performance of our system on different music styles and be robust to different annotation strategies.

RWC Classical [10]: 60 western classical pieces, from 1 to 10 minutes. We removed the last track as the annotation seemed inconsistent.

RWC Jazz [10]: 50 jazz tracks from 2 to 7 minutes.

RWC music genre [11]: 92 music tracks from various music styles, from 1 to 10 minutes. We removed the traditional Japanese songs and the a Capella song as we don't have the corresponding audio.

RWC Pop [10]: 80 Japanese Pop music and 20 American Pop music tracks from 3 to 6 minutes.

Beatles [3] **:** 179 songs from The Beatles.

Ballroom [4] **:** 698 30-second long excerpts from various ballroom dance music.

Hainsworth [12]: 222 excerpts from 30 second to 1 minute from various music styles. It is to note that the current downbeat annotation can significantly be improved.

Klapuri subset [15]: The downbeat annotations for this dataset are lacking in some files. Full cleaning will be done in future work but we use a subset of 4 relatively difficult genres for downbeat tracking : Jazz, Electronic music, Classical and Blues with 10 randomly selected excerpts for each genre.

Quaero [5] **:** 70 songs from various Pop, Rap and Electronic music hits.

4.3 General train/test procedure

We use a leave-one-dataset-out approach, meaning that we train and validate our system on all but one dataset and test it on the remaining one. Compared to standard cross-validation, this procedure was chosen to be more fair to non machine learning methods that are blind to the test set and to supervised algorithms using the same approach. However, it is limiting the ability of the deep networks and the CRF model to work on test data from styles not often seen in the training set. Two notable examples are the RWC Classical and RWC Jazz music datasets.

Audio_Downbeat_Estimation

[3] http://isophonics.net/datasets

[4] http://www.ballroomdancers.com

[5] http://www.quaero.org

Dataset	ll	ll + da	pl	pl + da
RWC Jazz	65.3	66.0	65.5	**66.1**
RWC Class	44.3	44.3	43.8	**45.9**
Hainsworth	62.9	65.9	64.5	**66.0**
RWC Genre	66.2	68.1	69.1	**69.3**
Klapuri subset	67.1	71.2	67.4	**71.5**
Ballroom	78.0	77.3	79.0	**80.9**
Quaero	83.5	**83.8**	83.1	82.7
Beatles	84.0	84.1	84.4	**85.2**
RWC Pop	87.2	85.1	86.7	**87.4**
Mean	70.9	71.8	71.5	**72.8**

Table 1. F-measure results for different configurations of the presented system. ll means the features come from the network last layer and pl means that features from the penultimate layer were also used. da means data augmentation was used.

4.4 CRF training

For CRF training we use the Pycrfsuite toolbox [21]. The CRF parameters are learned as classically done in a maximum likelihood sense using both ℓ_2 and ℓ_1-regularisation, thus in an elastic-net fashion, so as to promote sparse solutions, and solved for using the L-BFGS algorithm. The optimal values of the regularisation parameters were selected by a 4-fold cross-validation on the training set. For the last layer features, the grid for the optimal ℓ_2 value is [100,10,1.0,0.1,0.01,0.001,0.0001]. Since there are only four features out of the networks, we don't need feature selection and the ℓ_1 parameter was set to 0. When adding the penultimate layer features, the grids for the optimal ℓ_1 and ℓ_2 values were [10,100] and [0.1,0.01,0.001] respectively.

5. RESULTS AND DISCUSSION

5.1 Impact of the data augmentation:

Configuration using data augmentation will be abbreviated by "da", and their F-measure results for each dataset is shown in table 1. We can see an improvement on all datasets, except on the RWC Pop and Quaero datasets. Indeed, the number of songs containing 3 or 6 tatums per bar is very limited there. Overall the F-measure improvement is of +0.9 percent point using last layer features (abbreviated by "ll") and of +1.3 pp using penultimate layer features (abbreviated by "pl").

5.2 Impact of the penultimate layer:

F-measure results of configurations adding the penultimate layer output as features is also shown in table 1. Using the penultimate layer increases the results overall by 0.6 pp with the non augmented data and by 1.0 pp with the

Dataset	[23]	[22]	[3]	[17]	[6]	pl + da
RWC Jazz	39.6	47.2	42.1	51.5	**70.9**	66.1
RWC Class	29.9	21.6	32.7	33.5	**51.0**	45.9
Hainsworth	42.3	47.5	44.2	51.7	65.0	**66.0**
RWC Genre	43.2	50.4	49.3	47.9	66.1	**69.3**
Klapuri	47.3	41.8	41.0	50.0	67.4	**71.5**
Ballroom	45.5	50.3	50.0	52.5	80.1	**80.9**
Quaero	57.2	69.1	69.3	71.3	81.2	**82.7**
Beatles	53.3	66.1	65.3	72.1	83.8	**85.2**
RWC Pop	69.8	71.0	75.8	72.1	**87.6**	87.4
Mean	47.6	51.7	52.2	55.8	72.6	**72.8**

Table 2. F-measure results for compared algorithms. [23], [22] and [3] are unsupervised. [17] and [6] are supervised algorithms also trained with a leave-one-dataset-out approach. [6] uses the same training sets and [17] uses similar training sets, with the addition of the Boeck [1, 2], Rock [28] and Robbie Williams [8] datasets and the subtraction of the Klapuri subset and the Quaero dataset.

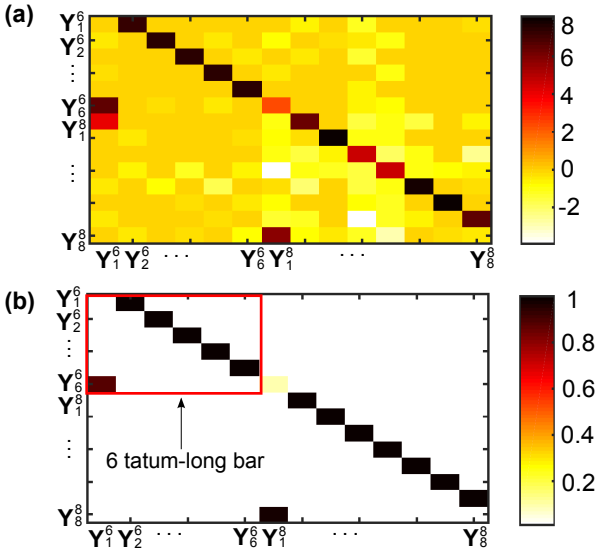

Figure 3. Selected transition weights for the 6 and 8 tatum-long bars. It corresponds to the output labels Y_1^6 to Y_6^6 and Y_1^8 to Y_8^8. (a) Weight of the transition feature function in the presented CRF model. (b) Coefficients of the transition matrix in [6]. As an illustration, inside the red rectangle pointed by an arrow are all the coefficients corresponding to a transition inside a 6 tatum-long bar. There is a weight at the bottom left corner of this rectangle with a value close to 1. It corresponds to the weight of the transition from Y_6^6 to Y_1^6.

augmented data. Its impact on the bigger datasets (Ballroom, Beatles, RWC Pop, RWC Genre, Hainsworth), representing 85% of the songs is more important than for the smaller datasets. Besides, using both the data augmentation and the penultimate layer allows the CRF model to have the best performance on all datasets but one, and to have a statistically significant improvement over the other configurations.

5.3 Comparison to other algorithms:

We compared our best system to the ones of Krebs et al. [17], Peeters et al. [23], Davies et al. [3] and Papadopoulos et al. [22]. We also compared to a system using the same neural networks but with a different feature combination and temporal model [6]. In this system, the output of the four networks is averaged and a Viterbi model with hand-crafted transition and emission probabilities is used to decode the downbeat sequence. Due to space constraints, we do not add [4] and [5] since they are close to [6] in terms of architecture and produce worse results. Results are shown in the table 2. With the new CRF system proposed here, the improvement is substantially better in all datasets compared to [17], [23], [3] and [22]. While the improvement averaged across datasets is moderate compared to [6], we observe a statistically significant improvement [6] . Overall results are held back by the performance on RWC Classical and RWC Jazz. The used training sets barely contain these music styles while we are exploiting a fully data-driven approach. The leave-one-dataset-out approach might be too restrictive when dealing with very distinctive music datasets. However, when more appropriate training data is available, the CRF model has a better potential, as results on RWC Genre indicates. This dataset includes more than 30% of Jazz and Classical music songs and has a significantly better performance with the new temporal model (69.3% F-measure compared to 66.1% in [6]). In this case the RWC Jazz and RWC Classical datasets were part of the training set and the CRF system was able to model these styles more accurately. In fact, the performance on Classical and Jazz music pieces on RWC Genre is improved by 6.8 pp, which is even better than the 3.1 pp overall. It highlights the potential of the data-driven proposed system, where relevant annotated data has a big impact on performance.

5.4 Analysis of the transition features:

The output space being similar with the one defined in [6], we can compare the transition coefficients. Due to space constraints, we limit our analysis to bars of 6 and 8 tatum of the $pl + da$ CRF model. They correspond to the most common bars in the used datasets. The transition coefficients can be seen in figure 3. The first observation is that the general intuition of moving circularly inside a bar is indeed learned by the CRF model as seen with the stronger weights of the transition feature function close to the diagonal of the figure. We also see that the proposed learned CRF model transition coefficients are more detailed while they seem more binary in [6]. The proposed system is less restrictive in metrical changes as can be seen by the coefficient of the output transitions $Y_6^6 \rightarrow Y_1^8$ and $Y_8^8 \rightarrow Y_1^6$ in particular. It can be because the observation features are

[6] It is to note that the comparison between the data augmented system and [6] is fair since the networks were trained on the same data, and the feature combination and temporal model steps of the heuristic model is blind to any data.

I_{HCNN}	I_{RCNN}	I_{MCNN}	I_{BCNN}
11.3	42.1	9.0	37.7

Table 3. Mean impact of each feature representation on the pl + da CRF model.

reliable enough to avoid false metrical changes between 6 and 8 tatum-long bars. Finally, we see that some transitions have strong negative weights in the CRF model. $Y_4^8 \rightarrow Y_1^8$ and $Y_7^8 \rightarrow Y_5^8$ have the top negative weights, both at -3.9. In the first case, it corresponds to going back to the downbeat after 4 tatums and in the second case to going back to the downbeat after 16 tatums while being in a 8 tatum-long bar. Finding the difference between a 4, 8 and 16 tatum-long bar is indeed quite difficult perceptively and for the networks. There can be one part of the song where the chords or the rhythmic patterns change twice as fast or twice as slow, which could misled the observation features. The negative weights can therefore emphasize a metrical continuity in the decoding.

5.5 Ability to find the correct metrical level:

To evaluate the ability of the system to find the correct metrical level, we use the continuity-based metric focusing on the total proportion of correct regions at the correct metrical level (CMLt) with a tolerance window of $\pm 17.5\%$ of the inter-beat-interval [7]. The proposed system obtains a CMLt of 61.5% while [6] obtains a CMLt of 56.6%. The CRF model is therefore more efficient to find the correct metrical level compared to [6]. It can be explained by the fact that every downbeat and non downbeat outputs have a different observation features while all the non downbeat states and all the downbeat states had the same observation feature respectively in the compared system. Besides, as seen above, the transition coefficients of the CRF model are better to avoid octave errors on duple meters while the compared system makes more errors there.

5.6 Analysis of the selected features:

We looked at the weight of the $pl + da$ CRF model to see if a feature representation had more impact than others to detect the downbeat sequence. To do so we calculated the sum of the absolute learned weight value belonging to each feature representation:

$$I_{XCNN} = \sum_{j \in XCNN} |\theta_j| \quad (2)$$

with $X \in \{H, R, M, B\}$. Results are shown in table 3 after a normalization inside and across datasets. It is to note that they are consistent for each dataset and each label. We can see that the rhythmic and bass content networks have a larger impact on the CRF model. It can be surprising knowing that the harmonic network is the best performing network in [6]. However, the rhythmic and bass content networks were trained to recognize the downbeat sequence on the whole input and not a single downbeat per input only. It allows them to encode information about the metrical level that is useful for the CRF model.

[7] We don't consider $\pm 17.5\%$ of the inter-downbeat-interval since it would be too permissive.

6. CONCLUSIONS

We presented a Conditional Random Field system based on multiple deep learned feature representations for the task of downbeat tracking. Using the networks penultimate layer feature representation with 3 beats per bar augmented data, we outperformed 5 compared downbeat tracking algorithms overall. While we need the training and test data to come from similar music styles to make full use of our powerful temporal model, it holds more potential compared to heuristic based approaches and could be more easily adapted to different music styles.

Future work will focus on learning the deep networks and the conditional random field models jointly and on refining the initial temporal segmentation.

7. REFERENCES

[1] J. P. Bello and J. Pickens. A robust mid-level representation for harmonic content in music signals. volume 19, 2005.

[2] S. Böck, F. Krebs, and M. Schedl. Evaluating the online capabilities of onset detection methods. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, 2012.

[3] M. E. P Davies and M. D. Plumbley. A spectral difference approach to extracting downbeats in musical audio. In *Proceedings of the European Signal Processing Conference (EUSIPCO)*, 2006.

[4] S. Durand, J. P. Bello, B. David, and G. Richard. Downbeat tracking with multiple features and deep neural networks. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[5] S. Durand, J. P. Bello, B. David, and G. Richard. Feature adapted convolutional neural networks for downbeat tracking. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

[6] S. Durand, J. P. Bello, B. David, and G. Richard. Robust downbeat tracking using an ensemble of convolutional networks. *arXiv preprint arXiv:1605.08396*, 2016.

[7] T. Fillon, C. Joder, S. Durand, and S. Essid. A conditional random field system for beat tracking. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[8] B. D. Giorgi, M. Zanoni, A. Sarti, and S. Tubaro. Automatic chord recognition based on the probabilistic modeling of diatonic modal harmony. In *Proceedings*

of the International Workshop on Multidimensional Systems (nDS), 2013.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 580–587, 2014.

[10] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: Popular, classical and jazz music databases. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, volume 2, pages 287–288, 2002.

[11] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: Music genre database and musical instrument sound database. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, volume 3, pages 229–230, 2003.

[12] S. Hainsworth and M. D. Macleod. Particle filtering applied to musical tempo tracking. *EURASIP Journal on Applied Signal Processing*, 2004:2385–2395, 2004.

[13] A. Holzapfel, F. Krebs, and A. Srinivasamurthy. Tracking the "odd": Meter inference in a culturally diverse music corpus. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 425–430, 2014.

[14] T. Jehan. Downbeat prediction by listening and learning. In *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 267–270, 2005.

[15] A. Klapuri, A. Eronen, and J. Astola. Analysis of the meter of acoustic musical signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(1):342–355, 2006.

[16] P. Korzeniowski, S. Böck, and G. Widmer. Probabilistic extraction of beat positions from a beat activation function. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, 2014.

[17] F. Krebs, S. Böck, and G. Widmer. An efficient state-space model for joint tempo and meter tracking. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, pages 72–78, 2015.

[18] F. Krebs, A. Holzapfel, A. T. Cemgil, and G. Widmer. Inferring metrical structure in music using particle filters. *IEEE Transactions on Audio, Speech and Language Processing*, 23(5):817–827, 2015.

[19] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields : Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICLM*, 2001.

[20] B. McFee, E.J. Humphrey, and J.P. Bello. A software framework for musical data augmentation. In *Proceedings of the International Conference on Music Information Retrieval (ISMIR)*, 2015.

[21] N. Okazaki. Crfsuite: a fast implementation of conditional random fields (crfs), 2007.

[22] H. Papadopoulos and G. Peeters. Joint estimation of chords and downbeats from an audio signal. *IEEE Transactions on Audio, Speech and Language Processing*, 19(1):138–152, 2011.

[23] G. Peeters and H. Papadopoulos. Simultaneous beat and downbeat-tracking using a probabilistic framework: Theory and large-scale evaluation. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(6), 2011.

[24] J. Peng, L. Bo, and J. Xu. Conditional neural fields. In *Advances in neural information processing systems*, pages 1419–1427, 2009.

[25] A. Srinivasamurthy, A. Holzapfel, and X. Serra. In search of automatic rhythm analysis methods for turkish and indian art music. *Journal of New Music Research*, 43(1):94–114, 2014.

[26] C. Sutton and A. McCallum. Dynamic Conditional Random Fields : Factorized Probabilistic Models for Labeling and Segmenting Sequence Data. In *Proceedings of ICML*, 2004.

[27] C. Sutton and A. McCallum. *An introduction to Conditional Random Fields for relational learning*, chapter 4, pages 93–128. MIT Press, 2006.

[28] D. Temperley and T. d. Clercq. Statistical analysis of harmony and melody in rock music. *Journal of New Music Research*, 42(3):187–204, 2013.

[29] J. Thomassen. Melodic accent: Experiments and a tentative model. *Journal of the Acoustical Society of America*, 71:1596, 1982.