Deep Learning for Automatic Downbeat Tracking

Dale Luginbuhl
Department of Computer Science
University of Cincinnati
Cincinnati, USA
luginbdr@mail.uc.edu

Atharva Pingale
Department of Computer Science
University of Cincinnati
Cincinnati, USA
pingalac@mail.uc.edu

Rithvik Reddy Sama
Department of Computer Science
University of Cincinnati
Cincinnati, USA
samary@mail.uc.edu

Lewis Thelen
Department of Computer Science
University of Cincinnati
Cincinnati, USA
thelenlr@mail.uc.edu

***Abstract*—Automatic downbeat detection is an important fundamental problem in music information retrieval. Successfully detecting downbeats in an excerpt of music enables solving higher-level music information retrieval tasks, as well as a host of end-user applications. Current state of the art models use powerful transformer architectures, which call for large amounts of data to successfully train. In a problem space that lacks large amounts of labeled data, we theorize that performing data augmentations to increase the amount of labeled data available will yield better results when using these powerful state of the art models. Constrained by both access to existing datasets, as well as access to compute we were forced to train on a subset of available data. As a result, even with data augmentations, we faced challenges with over-fitting. The data augmentations did not yield better results than the current state of the art model. With full access to available datasets, and additional compute resources we may see improved results.**

***Index Terms*—music downbeat tracking, music information retrieval, deep learning**

I. Introduction

By endowing computers with the ability to listen to music we unlock a number of applications for computer automation in music that were previously unavailable. This field of study is referred to as Music Information Retrieval (MIR). MIR is broken down into a number of overlapping problems as described in [1]. Examples include:

- Beat and downbeat tracking
- Tempo, time signature, and key estimation
- Melody tracking
- Chord detection
- Music structure analysis
- Musical onset/offset detection
- Mood or emotion recognition

We explore one of these problems, downbeat tracking, in more detail. Music is typically organized temporally around a rhythmic pulse. Each occurrence of the pulse is referred to as a *beat*. The goal of beat tracking is to annotate the locations of these beats in an excerpt of music. Beats can further be organized into *measures*, where each measure contains the same fixed number of beats. The first beat of each bar is typically emphasized, and this beat is known as the *downbeat*. The number of beats per measure, and the emphasis on the downbeat give a piece of music a certain feel, and are important qualities that shape how we interpret the music. The goal of downbeat tracking is to annotate the location of downbeats in an excerpt of music.

Successfully annotating the location of downbeats is important for two reasons. One, it can be used directly in applications. For example it could be used to analyze the performance of a practicing musician and provide feedback on how well they were able to maintain a consistent rhythmic pulse. Or it could be used to create sync points to allow an editor to easily synchronize a video or other multimedia application to the music. The second reason that downbeat tracking is important, is that it can be used as a pre-processing step in solving higher-level MIR tasks. If we have annotations of downbeat locations in an excerpt of music, these locations provide us with a temporal framework of the excerpt. This can simplify tasks like music structure analysis, melody tracking, or musical onset/offset detection, because these events typically occur on or around these beat and downbeat locations.

Downbeat tracking is a challenge because across different styles of music there are not consistent rhythmic or harmonic patterns that can be used to identify the first beat in a measure of music. Current methods use transformer-based architectures to take advantage of the attention mechanism. These methods apply to transformers to both the temporal and spectral domain to jointly capture information from these sequences.

Contribution. Our contribution is to enhance the limited datasets used for downbeat and beat detection by performing various data augmentations. We also experiment with using mel-scaled spectrograms during pre-processing rather than the usual power spectrograms.

II. Related Work

A. Datasets

Traditionally beat and downbeat tracking are assessed on a collection of 9 different datasets. Specifically these are the Ballroom dataset made up of albums of a variety of ballroom dance genres with a duration of 05 hours 57 minutes, the Harmonix dataset made up of individual tracks pulled from Western pop music with a duration of 56 hours, SMC dataset has 02 hours 25 minutes of audio tracks though little description exists of its contents, RWC-POP dataset has 06 hours 47 minutes of contemporary pop music, Beatles dataset has 08 hours 09 minutes comprised of the entire discography of The Beatles, the Hainsworth dataset has 03 hours 19 minutes of audio files, Simac dataset,with 03 hours 18 minutes audio files, HJDB dataset has 03 hours 19 minutes of audio files comprised of electronic dance music genres such as Hardcore, Jungle and Drum and Bass and lastly the GTZAN dataset has 08

hours 20 minutes across a variety of Western genres including blues, reggae, disco and heavy metal. Each dataset contains the data itself generally in the form of raw audio .wav files as well as the target. In our case the target annotations are time stamps in seconds of the location of the beats within the audio track as well as a corresponding label of the type of beat (ie a 1 for the downbeat, 2 for the next beat and so on up to 4). An example audio file from the Ballroom dataset with only the downbeat labels shown is given below. While there is no standard dataset or combination of datasets within the MIR beat tracking community it is generally common to see a model trained and tested on some combination of the above datasets.

B. Models

Previous models have employed a wide variety of strategies in both beat and downbeat tracking (we include mention of beat tracking only because beat tracking techniques often generalize to the special case of downbeat tracking or inadvertently occur as an intermediary step in the process of downbeat tracking). Earlier models [2], [3], [4] relied heavily on feature engineering to extract relevant events from the music (ie chord changes, accents etc.) in order to identify beat patterns within the audio file. More recent models have taken advantage of "data-driven" approaches, made possible by the rise of deep learning techniques, which allow for the model itself to learn and then extract relevant features from the raw audio file.

Unsurprisingly, of the deep learning approaches used in downbeat tracking, sequential models have found the best results. Earlier sequential models have used more vanilla sequential techniques such as Long Short Term Memory (LSTM) applied directly to the raw audio data as in the case of [5] others have found success with more complicated approaches such as Convolutional Recurrent Neural Networks (C-RNNs) applied to spectrogram representations of the audio data as in the case of [6]. However, such models suffer from issues with vanishing gradient when faced with longer music segments as is common with recurrent models operating on time series data. More contemporary models have tried to overcome these issues by employing the then novel Temporal Convolutional Network (TCN) architecture most notably in the case of [7]. The model employed convolutional filters across the temporal dimension of the audio file and achieved what was until recently state of the art performance.

Current state of the art models have taken advantage of recent advances with transformer based architectures to achieve even higher performance at downbeat tracking as well as more general MIR tasks such as chord change detection and music tagging than previous RNN and TCN models. Within transformer based downbeat tracking two divergent strategies appear to have emerged. The first engages in more explicit feature extraction as in the case of [8] where the data from individual instruments within the audio track are extracted and then fed together into a multi-headed attention transformer. This approach is able to achieve impressive performance results but struggles with generalizing beyond particular genres. Attention heads become accustomed to relations between particular instruments and so will struggle when encountering different arrangements. Consider the difference between a rock band and a symphony orchestra. Perhaps an explosion in the number of attention heads could be used to solve this problem, but questions of feasibility and efficiency then begin to arise. In sum this first strategy struggles with similar challenges that other feature engineering approaches struggle with.

The second strategy pushes further with the "data-driven" approach by relying on the model architecture to extract relevant features from the raw audio as in the case of [9] where novel Transformer in Transformer (TNT) blocks were utilized to mutually inform both spectral level and temporal level analysis of a raw audio file. These blocks were accompanied with a pre-processing module that work to convert the incoming audio into a spectrogram and then pass it through a residual network to extra relevant features before passing it off to the transformer portion of the network. The pre-processing is trained as part of the network training and so the network is effectively in control of the feature engineering. This SpecTNT model currently represents the state of the art in the combined task of beat and downbeat detection. We take this SpecTNT model to be our starting point for improvement.

Performance on the state of the art was slightly boosted through a combined SpecTNT-TCN network, however the model itself was hardly innovative. In effect the model worked by using a shared pre-processing module with the then embedded data fed into both a TCN and SpecTNT model in parallel with the resulting output averaged to produce the combined network's final prediction. However, what little improvement is achieved comes at the cost of roughly twice the compute and does little to actually advance the methodology. In contrast our current study aims to push the performance of the SpecTNT model by following the data-driven/feature-engineered dialectic. SpecTNT represents the current cutting edge of the data-driven approach but this does not mean it could not be improved through more human led feature engineering. As a result we aim to improve the performance of SpecTNT along this route.

C. Challenges

Beat and downbeat tracking in music analysis pose several significant challenges. Early heuristic-based methods faced limitations in accurately detecting beats, relying on manually encoded rules that may not generalize well across diverse musical styles. Data-driven approaches, while effective, require substantial annotated data for training, which can be both costly and limited. Downbeat tracking is a challenging task because it often relies on other sub tasks such as beat tracking, tempo and time signature estimation and also because of the difficulty to state an unambiguous ground truth [8] [18]. Additionally, incorporating local spectral information and facilitating the exchange of critical local details across disparate temporal positions are formidable tasks. Both beat and downbeat tracking also grapple with the scarcity of annotated data, a prevalent issue in training data-intensive models like

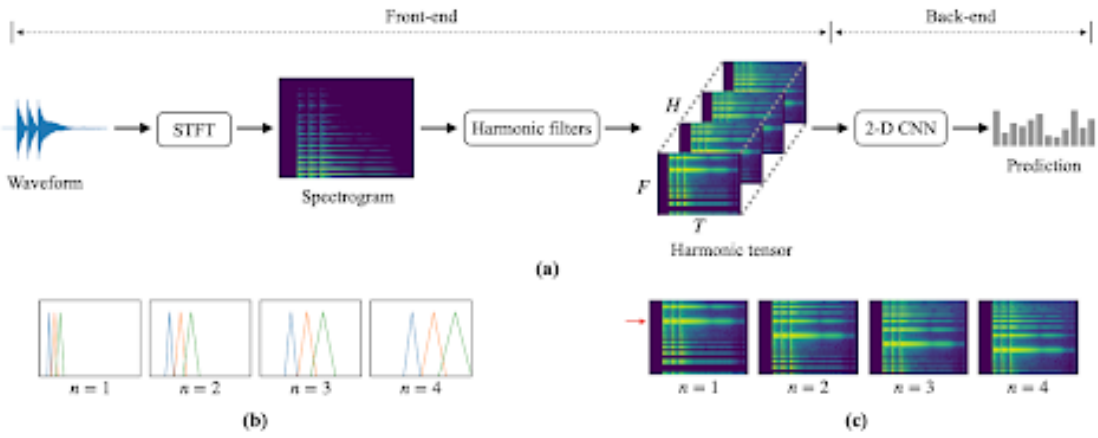

Fig. 1. A block diagram of the data-driven harmonic filters used as the front-end of the model in order to extract a more information-rich representation of the audio sample, as provided in [19].

Transformers [8] [14]. The generalization problem of deep learning models, which may fail to adapt to unseen or novel music pieces with different time signatures, tempo ranges, or musical characteristics than the training data [8].

III. Method and Dataset

A. Dataset

Due to limited computational power availability and some datasets being unavailable for public use, we decided to use only one dataset i.e. Ballroom. The Ballroom dataset comprises segments extracted from 698 music compositions, each approximately 30 seconds in duration. The total duration is 5h 57m. The dataset encompasses eight distinct sub-genres within ballroom dance music: Jive, Quickstep, Tango, Waltz, Viennese Waltz, Samba, Cha Cha Cha, and Rumba. [8] The labels for the dataset are timestamps (in seconds) at downbeat location as shown in Fig. 3. We initially proposed 60-20-20 split but due to limited data, we decided to proceed with higher split for training. The dataset is divided as 80% for training, 10% for validation and 10% for testing.

B. Method

The current state-of-the-art (SOTA) architecture uses a data-driven harmonic filter [19] as a ”front-end” in order to extract relevant features from the audio sample, and then a SpecTNT [9] architecture as a ”back-end” used to extract the desired labels. The input audio samples are a sequence of amplitude readings representing a discrete sampling of an audio waveform. This provides temporal information about the audio, but lacks additional information about the spectral or harmonic content of the audio. This is important information relevant to downbeat detection, because often times the spectral and harmonic content of the audio changes on downbeats. The front-end of the model addresses these concerns and extracts a new representation of the audio that contains explicit temporal, spectral, and harmonic content.

As shown in 1 the first step of the front-end is to perform a Short-Time Fourier Transform (STFT) to produce a spectrogram representation of the audio clip. We used a window size of 512 samples, and a hop length which refers to the number of samples between consecutive frames or windows in a time-domain signal is initialized to 256. The STFT looks at the window of 512 samples and converts this into a tensor of frequency content. Each index in the tensor represents a frequency in hertz (Hz), and the element at that index represents the magnitude of that frequency within the window of audio samples. This frequency tensor is referred to as a ”frame”. Now the STFT algorithm shifts the window by the hop length (256 in our case), and produces another frame. The algorithm repeats this for the full sequence of input audio samples. The result is a 2-D tensor where one dimension is frames, which represents the temporal domain, and the other dimension is frequency, which represents the spectral domain.

Now we apply a filter bank of learnable bandpass filters to the spectrogram, resulting in a series of spectrograms, each corresponding to one of the applied filters. A bandpass filter is parameterized by a ”center frequency”. When the filter is applied to a spectrogram it allows the frequencies around the filter’s center frequency to pass through, while attenuating the frequencies outside of the filters frequency band. What this looks like in the 2-D tensor representing the spectrogram is that the magnitudes of the frequencies in the frequency dimension that fall within the bandpass filter are left unchanged, while the the magnitudes of the frequencies outside the bandpass filter are attenuated. A series of these bandpass filters, each with a different center frequency, is referred to as a filter bank. This filter bank, when applied to a spectrogram, results in a new spectrogram for each filter in the bank. Each spectrogram in the resulting series highlights a different frequency of the audio sample, representing a fundamental frequency and it’s harmonic series. This new series of spectrogram constitutes a third dimension to our data representation, representing the harmonic content. We now

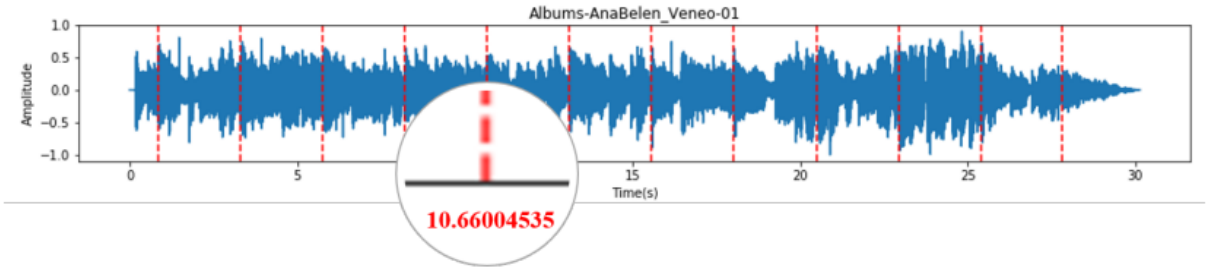

Fig. 2. Example of a typical downbeat annotation (Albums-AnaBelen Veneo-01.beat from Ballroom dataset), showing downbeat time (in red dashed line). [8]

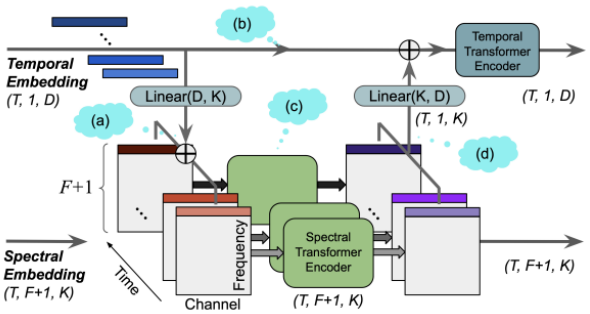

Fig. 3. The block diagram of a SpecTNT block. Tensors and modules are illustrated with non-rounded and rounded rectangles, respectively.

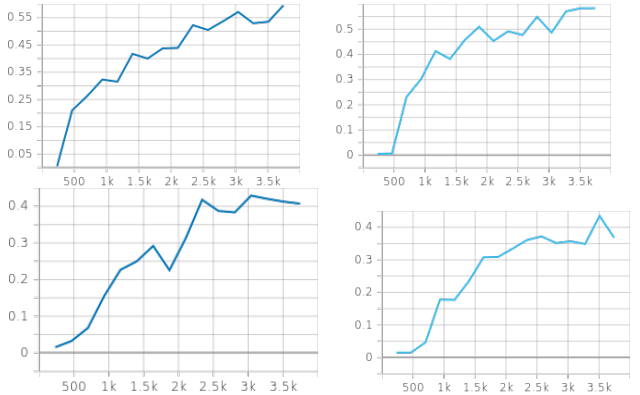

Fig. 4. Graph showing F Measure of Beat (top) and Downbeat (bottom) of Vanilla Spec-TNT and Augmented SpecTNT respectively across each Gradient step

have a 3-D tensor whose dimensions represent the temporal, spectral, and harmonic content of the input audio sample. This new representation of the audio can then be handed off to the backend SpecTNT architecture for processing.

The SpecTNT architecture consists of a convolutional module, positional encoding, SpecTNT module, and output module. An input harmonic representation is first processed by the front-end convolutional layers which is ResNet. Then, we stack 3 residual units, each uses 256 feature maps and a kernel size of 3. The main SpecTNT module is formed by stacking multiple SpecTNT blocks. A SpecTNT block consists of a spectral encoder and a temporal encoder. We use 64 feature maps with 4 attention heads for the spectral encoder to extract the spectral features into a set of frequency class tokens (FCTs) at each time step. Through the attention mechanism, a FCT can characterize useful harmonic and timbral components, which may better represent a chord or instrumentation. For the temporal encoder, we use 256 feature maps and 8 attention heads. It enables the important local spectral information to be exchangeable via FCTs to pay attention to the beat/downbeat positions. Finally, we stack 5 SpecTNT blocks, followed by a linear layer to output three activation functions for beat, downbeat, and non-beat.

In the context of enhancing the performance of SOTA model for downbeat, several crucial strategies have been employed. Due to the limited datasets with beat and downbeat annotations, data augmentation might be helpful in curbing overfitting. These augmentations serve to diversify the training data, thereby enhancing the model's adaptability to a broader range of musical contexts. Some of fairly common data augmentations are implemented, each of which has an associated probability p of being applied to each training example during a training epoch. This includes the applications of highpass and lowpass filters with random cutoff frequencies (p = 0.25), additive white noise (p = 0.05), applying a tanh nonlinearity (p = 0.2), shifting the beat locations forward or back by a random amount between ± 70 ms (p = 0.3), dropping a contiguous block of audio frames and beats of no more than 10% of the input (p = 0.05) [13]. In addition to data augmentation, we changed the spectrogram type from power spectrogram to mel-scaled spectrogram. Several researches state thet Mel spectrograms might be closer representations of how humans perceive music compared to note-based or raw waveform representations, and therefore imply that, using Mel spectrograms for audio related tasks would yield better results.

IV. Results

The results F Measure obtained for the model Vanilla SpecTNT for test Beat is 0.5449 and F measure for the SpecTNT using data augmentation is 0.5261. The Downbeat F Measure for the Vanilla SpecTNT is 0.4626 and for the proposed model it is 0.3796. The F Measure for Vanilla SpecTNT vs Augmentated Spec-TNT is given below

The result compare the F Measure of the beats for both the Vanilla SpecTNT and the Augmentated SpecTNT. The

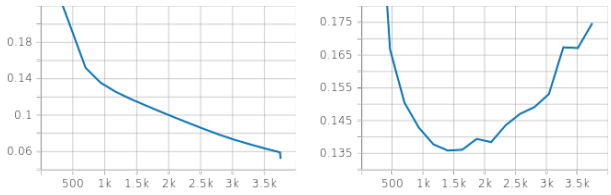

Fig. 5. Graph showing training loss and validation training loss for SpecTNT.

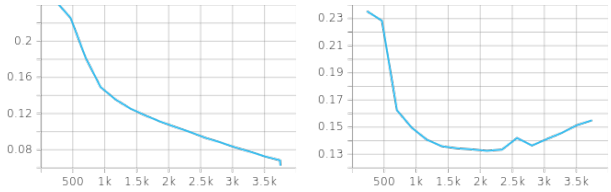

Fig. 6. Graph showing training loss and validation training loss for Augmented SpecTNT.

X-axis is initialized as the terms of gradient steps. At each gradient step the model weights are updated accordingly. For every 250steps which is roughly equals to 16batches are considered as one epoch. The graph given below depicts the loss obtained from the training and the validation loss for the Vanilla SpecTNT.

The loss is exponentially decreasing but for the validation set the loss decreased till 2500gradient step and eventually started increasing thus we can conclude the model is overfitting. The graph given below depicts the loss obtained from the training and the validation loss for the Augmented SpecTNT.

V. Discussion and Conclusion

The general strategy involves by increasing the dataset size and the complexity of the model, surpassing the capabilities of the state-of-the-art (SOTA) model. However, the pursuit of this strategy encounters a significant hurdle: the substantial volume of data is required for training the model. Most of the data, remained inaccessible as owners denied requests for access. In response, we implemented data augmentation techniques to expand the dataset.

Despite these efforts, the augmented data still fell short of meeting the requirements for effective training. Even with data augmentation in place, the insufficiency of training data persisted. The challenge of obtaining access to more data would amplify the temporal and computational costs associated with training the model. This scarcity of data and resources is reflected in the lower F-measure scores observed for both models. Despite these limitations, the attained results were remarkably impressive, even by utilizing only a fraction of the data compared to the SOTA model.

A. Future work

In order to advance automatic downbeat detection in music, future work should prioritize the expansion of datasets to encompass a broader spectrum of music genres and styles. This would enhance the model's ability to generalize across diverse musical contexts. Additionally, exploring innovative data augmentation strategies tailored to the unique characteristics of music data could further improve the model's adaptability. Investigating alternative model architectures or hybrid approaches may contribute to increased robustness and overall performance. Further attention to hyperparameter tuning and potential transfer learning from pre-trained models on larger music datasets could optimize the model's efficiency.

All team members have contributed in equal measure to this effort

References

[1] Choi, K., Fazekas, G., Cho, K., and Sandler, M., "A tutorial on deep learning for music information retrieval." arXiv preprint arXiv:1709.04396, 2017.

[2] Dixon, Simon (2001) Automatic Extraction of Tempo and Beat From Expressive Performances, Journal of New Music Research, 30:1, 39-58

[3] A. P. Klapuri, A. J. Eronen and J. T. Astola, "Analysis of the meter of acoustic musical signals," in IEEE Transactions on Audio, Speech, and Language Processing, vol. 14, no. 1, pp. 342-355, Jan. 2006

[4] Goto, Masataka. (2002). An Audio-based Real-time Beat Tracking System for Music With or Without Drum-sounds. Journal of New Music Research

[5] S. Böck, F. Krebs and G. Widmer, "Joint beat and downbeat tracking with recurrent neural networks", Proc. Int. Soc. Music Inf. Retrieval Conf., pp. 255-261, 2016

[6] Tian Cheng, Satoru Fukayama, Masataka Goto, "Joint Beat and Downbeat Tracking Based on CRNN Models and a Comparison of Using Different Context Ranges in Convolutional Layers", July 2021

[7] MatthewDavies, E. P., and Böck, S., "Temporal convolutional networks for musical audio beat tracking." In 2019 27th European Signal Processing Conference (EUSIPCO) (pp. 1-5). IEEE, September 2019.

[8] Zhao, J., Xia, G., and Wang, Y., "Beat Transformer: Demixed beat and downbeat tracking with dilated self-attention." arXiv preprint arXiv:2209.07140, 2022.

[9] Hung, Y. N., Wang, J. C., Song, X., Lu, W. T., and Won, M., "Modeling beats and downbeats with a time-frequency transformer." In ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 401-405). IEEE, May 2022.

[10] Jia, B., Lv, J., and Liu, D., "Deep learning-based automatic downbeat tracking: a brief review." Multimedia Systems, 25, 617-638, 2019.

[11] Han, K., Xiao, A., Wu, E., Guo, J., Xu, C., and Wang, Y., "Transformer in transformer." Advances in Neural Information Processing Systems, 34, 15908-15919, 2021.

[12] Devlin, J., Chang, M. W., Lee, K., and Toutanova, K., "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805, 2018.

[13] Steinmetz, C. J., and Reiss, J. D., "WaveBeat: End-to-end beat and downbeat tracking in the time domain." arXiv preprint arXiv:2110.01436, 2021.

[14] Chiu, C. Y., Ching, J., Hsiao, W. Y., Chen, Y. H., Su, A. W. Y., and Yang, Y. H., "Source separation-based data augmentation for improved joint beat and downbeat tracking." In 2021 29th European Signal Processing Conference (EUSIPCO) (pp. 391-395). IEEE, August 2021.

[15] Cheng, T., and Goto, M., "U-Beat: A Multi-Scale Beat Tracking Model Based on Wave-U-Net." In ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1-5). IEEE, June 2023.

[16] Di Giorgi, B., Mauch, M., and Levy, M., "Downbeat tracking with tempo-invariant convolutional neural networks." arXiv preprint arXiv:2102.02282, 2021.

[17] Hung, Y. N., Wang, J. C., Won, M., and Le, D., "Scaling Up Music Information Retrieval Training with Semi-Supervised Learning." arXiv preprint arXiv:2310.01353, 2023.

[18] S. Durand, J. P. Bello, B. David, and G. Richard, "Robust downbeat tracking using an ensemble of convolutional networks," IEEE/ACM Trans. Audio, Speech, and Language Process., vol. 25, no. 1, 2016.

[19] M. Won, S. Chun, O. Nieto, and X. Serrc, “Data-driven harmonic filters for audio representation learning,” In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 536-540). IEEE, May 2020.