ADAPTING METER TRACKING MODELS TO LATIN AMERICAN MUSIC

Lucas S. Maia[1] **Martín Rocamora**[2] **Luiz W. P. Biscainho**[1] **Magdalena Fuentes**[3]
[1] PEE/COPPE, Federal University of Rio de Janeiro, Brazil
[2] FING, Universidad de la República, Uruguay
[3] MARL-IDM, New York University, United States
lucas.maia@smt.ufrj.br

ABSTRACT

Beat and downbeat tracking models have improved significantly in recent years with the introduction of deep learning methods. However, despite these improvements, several challenges remain. Particularly, the adaptation of available models to underrepresented music traditions in MIR is usually synonymous with collecting and annotating large amounts of data, which is impractical and time-consuming. Transfer learning, data augmentation, and fine-tuning techniques have been used quite successfully in related tasks and are known to alleviate this bottleneck. Furthermore, when studying these music traditions, models are not required to generalize to multiple mainstream music genres but to perform well in more constrained, homogeneous conditions. In this work, we investigate simple yet effective strategies to adapt beat and downbeat tracking models to two different Latin American music traditions and analyze the feasibility of these adaptations in real-world applications concerning the data and computational requirements. Contrary to common belief, our findings show it is possible to achieve good performance by spending just a few minutes annotating a portion of the data and training a model in a standard CPU machine, with the precise amount of resources needed depending on the task and the complexity of the dataset.

1. INTRODUCTION

Meter tracking means following the pulsating temporal structure of music from audio signals, which implies identifying at least beats and downbeats [1]. It is a long-standing area of research in music information retrieval (MIR) with applications ranging from automatic DJ mixing [2] to musicological studies [3]. Meter tracking has gone through a big transformation in the last decade due to the introduction of deep learning (DL) techniques [4–7], which brought an improvement in performance as well as a change in the design paradigm of related methods [8]. Nowadays, beat and downbeat tracking models rely mostly on supervised DL [8], and thus become data-driven and requiring large amounts of annotated data to generalize to different songs, genres or datasets.

This dependence on annotated data poses many challenges to the widespread use and adoption of such models, especially for culturally specific music traditions [9–11], which often lack annotated data as producing annotations requires culturally-aware expertise. For this reason, off-the-shelf general-purpose models typically underperform in these music genres since they are underrepresented in the datasets used for training. Nevertheless, previous work on some Latin American music traditions shows that if annotations are available, training statistical models can produce good performance results [12].

Recent works have started to look at beat tracking from a different perspective. Instead of developing "universal" models capable of performing equally well across various music genres (requiring large quantities of labeled data), recent efforts have shifted towards adapting preexisting models to succeed on a subset of interest [13], which can be as restricted as a single musical piece [14, 15]. This paradigm aligns well with real-world applications, where it is reasonable for a user to spend a short time producing a few seconds of annotations to get a good performance.

We apply this idea to the refinement of a meter tracking model so that it works well in a particular music genre. We argue that if the genre presents enough homogeneity in terms of its instrumentation and metric structure, as is the case with many Latin American music traditions, it is possible to adapt meter tracking models to perform notably well with just a few annotated data points. We explore this adaptability in terms of data, performance and computational cost. While focusing on two Latin American music genres, *samba* and *candombe*, we study the adaptation of a deep learning state-of-the-art model [16] and compare it with a simpler statistical model [17]. Our contributions are: 1) We perform a detailed analysis on how much annotated data and computation time in CPU are needed to achieve close to "full-dataset" performance in *samba* and in *candombe*, including models trained from scratch and fine-tuned, and compare them to off-the-shelf models trained with Western music; 2) We propose initial experiments to understand the homogeneity conditions under which this adaptation will be successful; 3) We open-source our experiments and provide pre-trained models.

1.1 Other adaptive methods

Existing adaptive methods typically feature some form of transfer learning or fine-tuning, use deep learning models, and are concerned with either beat tracking [13–15] or onset detection [18]. Fiocchi et al. [13] adapt a beat tracking model via transfer learning from Western music to a dataset of Greek music. The authors explore both recurrent neural networks (RNNs) and long-short term memory networks (LSTMs), plus a dynamic Bayesian network (DBN) for inference. The model is fine-tuned using a big training set, though with limited success which the authors attribute to the challenges of the dataset. Even though this is an interesting approach for exploring the idea of adapting to a particular music genre, RNNs and LSTMs are known to be computationally expensive, so in the context of real-world model adaptation they are a concerning choice.

On the other hand, Pinto et al. [14] and Yamamoto [15] explore temporal convolutional networks (TCNs), and focus on the adaptation of models to a particular piece of interest of the user. These authors showed that is possible to adapt TCN models using very small quantities of data (in the order of seconds) to work well, in particular with musically challenging pieces. Furthermore, the TCN is a light-weighted model, computationally more efficient.

Finally, Fonseca et al. [18] apply similar ideas to the adaptation of an onset detection model (also featuring TCNs) to a Latin American music tradition: *maracatu de baque solto*. The authors fine-tune the last layers of the TCN with just a few seconds of manual annotations, and show the advantage of instrument-specific models for the automatic annotation of onsets for musicological studies.

1.2 Latin American Music Traditions

Candombe drumming is a musical tradition from Uruguay that constitutes an essential part of its popular culture and African heritage. Its rhythm is structured in 4/4 meter, and it is played while marching in the streets using three types of drums of different sizes and pitches: *chico*, *repique*, and *piano*. Each of these drums has a distinctive rhythmic pattern and musical role. An additional time–line pattern, called *clave* or *madera*, is shared by the three drums. The *chico* drum is the timekeeper; it repeats a one-beat pattern that establishes the pulse throughout the performance. The *repique* drum is the improviser; it alternates *clave* patterns and characteristically syncopated phrases. The *piano* drum delineates the timeline with distinctive one-cycle patterns and occasionally interposes ornamented *repique*-like figurations. The rhythm shares many traits with other musical traditions of the Afro-Atlantic world. Notably, some of its rhythmic patterns have strong phenomenological accents displaced with respect to the metric structure and divide the rhythmic cycle irregularly with few strokes on the beat.

In parallel, *samba* is a Brazilian musical genre deeply rooted in Brazilian culture, and also has African origins. The word "*samba*" actually describes a family of different subgenres, the most famous arguably being *samba de enredo*, *partido alto*, *bossa nova*, and *pagode*. Similarly to *candombe*, *samba* can be played while parading, which is most common during the festivities of *Carnaval*. It can also be performed in more informal settings, in *rodas* and bars, or even as a chamber-music-like style. Its rhythm is commonly perceived in 2/4 meter, and is conveyed by several types of percussion instruments —– *tamborim*, *pandeiro*, *surdo*, *cuíca*, *agogô*, among others. Each instrument has a handful of distinct patterns [19], and more than one instrument may act as the timekeeper. Because of this combination of timbres and pitches, the texture of a performance can become very complex. *Samba* has unmistakable characteristics as the strong accent on the second beat and the development of contrametric structures.

2. METHOD

Following our intuition about the high homogeneity of *candombe* and *samba* as discussed in Section 1.2, our objective is to understand if it is possible to train meter tracking models with small quantities of data from these music traditions, and if so, how much is needed. To that end, we train the models with increasing amounts of annotated data, ranging from less than a minute up to nearly 40 min, and compare the performance and computational cost of each configuration against the others. We contrast three different training strategies: 1) training the model from scratch with either *candombe* or *samba* snippets; 2) fine-tuning a model trained with 38 h of data from diverse datasets of Western music to work on either *candombe* or *samba*; and 3) same as the previous two, but training the models with data augmentation to artificially increase the "small data" input. We use a state-of-the-art temporal convolutional network model [16] for our experiments, as it presents a good compromise between performance and computational cost. We contrast this model against off-the-shelf models trained in Western music. To understand the adaptability and computational cost of deep learning based methods, we compare the TCN against another simple yet effective baseline, a Bayesian model (BayesBeat) [17]. In the following, we explain our methodology in detail.

2.1 Datasets

We have selected datasets of two different Afro-rooted Latin American music traditions for our experiments. First, the *Candombe* dataset [12, 20], which consists of 35 recordings of *candombe* drumming, for a total of nearly 2.5 h. Each track contains an ensemble recording of three to five drummers using different configurations of drums. Tempo varies greatly and often increases along the performance. To represent the *samba* genre, we use the "acoustic mixtures" data from the BRID dataset [21]. These correspond to 93 short tracks (about 30 s each) of musicians playing together rhythm patterns found in *samba* and two of its subgenres (*samba de enredo* and *partido alto*). Ten different instrument classes are represented, and two to four musicians take part in each track. The dataset contains a variety of tempi; however, tempo remains fairly constant within each track. In order to consistently train our models with about the same amount of data from both *candombe* and *samba*, and also to allow the comparison between the results obtained for both sets, *candombe* tracks were segmented into non-overlapping 30 s excerpts. In each experiment repetition, we use a sample of 93 *candombe* excerpts.

We also used six datasets to train the baseline TCN model: Ballroom [17, 22], Beatles [23], GTZAN [24, 25], and RWC (Classical, Popular, Jazz) [26, 27]. These are commonly used in meter tracking tasks and together correspond to over 38 h of audio data. The Ballroom and GTZAN datasets comprise many diverse music genres (e.g., waltz, tango, rumba, rock, pop, country, etc.). We used the loaders from *mirdata v0.3.6* [28], except for a custom loader used with Ballroom.

2.2 Working with small size datasets

For our experiments, in all cases, we first separate train and test data (80% and 20% of 93 excerpts respectively) to ensure a fair assessment of the models. Then, we divide the training data into six subsets, spanning {4, 9, 18, 37, 55, 74} 30-second tracks. We want to determine how differently the models adapt to small quantities of data, so we followed a similar approach to that of [14] to define the amount of data to be used for training. We select short 10 s temporal regions at the beginning of the audio excerpts, along with the corresponding beat and downbeat annotations, and discard the remaining audio portion. Then we split each of these regions into two adjacent 5 s parts, the first to be used for training and the second reserved for validation in the TCN model; alternatively, we use the entire 10 s for training the Bayesian model with off-the-shelf parameters. Considering that each snippet only lasts 10 s, these data subsets add up to approximately 40 s, 1.5, 3, 6, 9, and 12 min of annotations, respectively. The rationale behind this strategy is that given a set of recordings of such Latin American music traditions in real-world applications, it would be reasonable to ask a user to annotate just a few seconds to a few minutes of data; of course, the less data needed, the better.

Given that we are using very few data points to train the models, performance is strongly affected by data sampling. To mitigate this, we repeat all of our experiments 10 times with different seeds for the random data split generation, which means that models are trained 10 times with each of the different subset sizes. Note that selecting the best strategies for data sampling is out of the scope of this work, and left to be addressed in the future. Test data are left uncut, i.e., we use the full 30 s, to keep compatibility with common model evaluation practices in meter tracking.

2.3 TCN Model

We use in our experiments the TCN multi-task model presented in [16], in particular the open-source implementation of [8]. In this work, we focus on meter tracking, and ignore the tempo estimation head of the network. First, the TCN estimates the beat and downbeat likelihood. Then, we use two different implementations of a DBN (*DBNBeatTracker* and *DBNDownBeatTracker* from *madmom v0.17.dev0* [29]) to infer the final positions of beats and downbeats respectively. Inferring them separately rather than jointly led to better results.

2.4 Training strategies

2.4.1 Training from scratch (TCN-FS)

For datasets with high similarity in terms of instrumentation, rhythmic patterns, and tempo, we expect that we can train a model from scratch with a few training points that would work well for most of the data.

Following the explanation in Section 2.2, we train one model per data subset, and repeat this 10 times with randomly initialized weights and seeds. We also consider the case in which all annotations are available and include the analysis of model performance when training with the entire 30-second excerpts. In this situation, we split the 74 train excerpts into train and validation (75%/25%). For every strategy, we use a learning rate of 0.005, and reduce it by a factor of 0.2 if validation loss did not improve after 10 epochs. We train for a maximum of 100 epochs, early stopping at 20 epochs.

2.4.2 Fine-tuning (TCN-FT)

We also approach the problem of meter tracking in a culture-specific setting from a “transfer learning” perspective. Following [13, 14, 18], we adapt a meter tracking model that was previously trained for a different musical context. The intuition here is that if the model is first trained on a large dataset, even if it was built around Western music, it can serve as a good starting point for a model that is to be tuned for a specific out-of-training music tradition. This is a realistic approach since most of the available annotated data and trained models are Western-based. For this purpose, we trained a baseline TCN model on the Ballroom, Beatles, GTZAN, and RWC datasets. Due to the nature of its training data, this baseline model has to cope with many different meters, genres and acoustic conditions, which makes it a good starting point. We fine-tuned it by using the same training procedure described previously with the initial learning rate reduced to 0.001, a fifth of the value used in the FS approach, as in [14].

2.4.3 Data augmentation (TCN-FTA, TCN-FSA)

Data augmentation techniques are useful for artificially increasing the number of training data points, which can be of great benefit in cases of low or insufficient data such as ours. In order to evaluate the impact of data augmentation in our models, we adopted a simple strategy inspired by [14, 16] in the experiments conducted with the TCN model: computing the input STFTs with different frame rates, i.e., varying hop sizes, so as to even out the distribution of tempi in the train set. Instead of randomly sampling from a normal distribution around the annotated tempo, we selected a set of frames rates ±2.5% and ±5% around its value. This allowed us to increase our sample size five-fold while maintaining the same amount of annotation effort. Models obtained with the data augmentation procedure are labeled TCN-FSA and TCN-FTA, for the training strategies described in Sections 2.4.1 and 2.4.2.

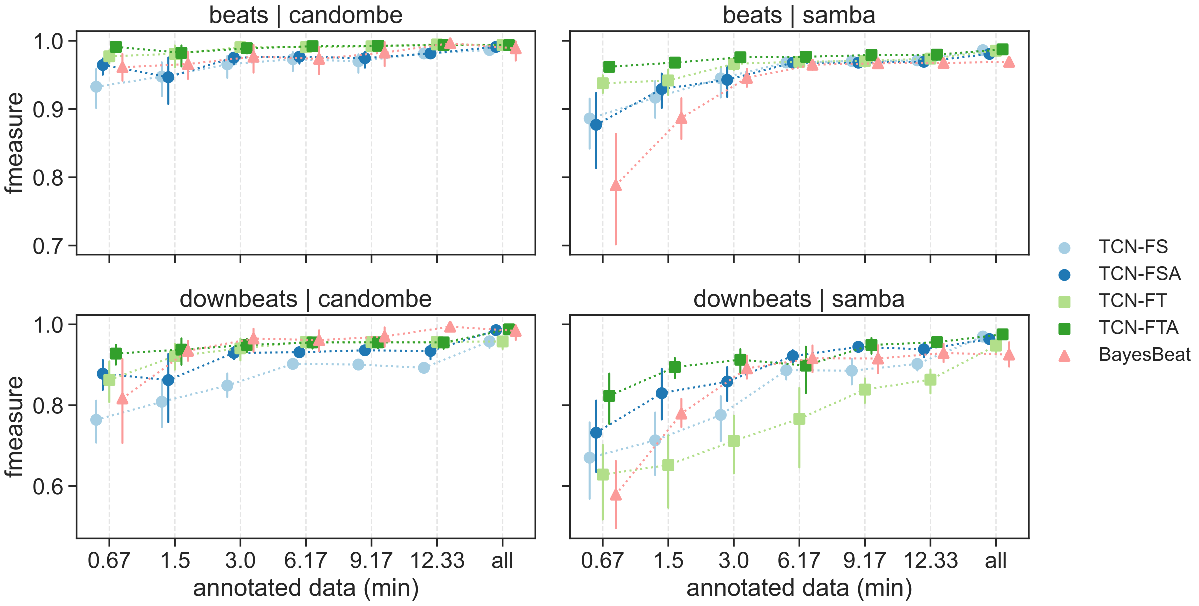

Figure 1: Performance of different model and training configurations. Label “all” indicates fully-annotated dataset.

2.5 Baselines

We include two types of baselines. Firstly, the BayesBeat statistical model [17] is used as reference to the adaptability and computational cost of the TCN. It has fewer parameters, thus training is faster. The second type of baselines are three off-the-shelf models—a Signal Processing technique, and two neural networks trained on Western music—; they illustrate the need for tailor-made solutions/adaptations in our context. Details presented below.

BayesBeat. This is based on the *dynamic bar pointer model* [30], and it simultaneously estimates beats, downbeats, tempo, meter, and rhythmic patterns, by expressing them as hidden variables in a hidden Markov model (HMM). An observation feature based on the spectral flux is computed from the audio signal and an observation model uses Gaussian mixture models (GMM) that are fitted during training to the feature values of each bin in a one–bar grid, so that rhythmic patterns are learned. Several patterns can be modeled, though one pattern is assumed to remain constant throughout the audio signal.

BayesBeat has a few hyperparameters that the user should choose depending on the music. Those are the number of rhythmic patterns, the type of feature to use (e.g., using only low, or low and high frequencies), and the feature grouping (e.g., how to compute the rhythmic pattern clusters), the tempo range, and whole note subdivisions. In [17], it is reported that using two separate frequency bands ($\gtrless$ 250 Hz) helps finding the correct metrical level and is beneficial for beat and downbeat tracking. But, considering more frequency bands did not improve the results [17]. According to [31], using one rhythmic pattern per rhythm class is usually enough to achieve a good performance and provides the best results in most cases. Following this, we use one rhythmic pattern and two frequency bands.

Off-the-shelf baselines. We use the joint beat and downbeat tracking model of Böck et al. [4] as per its implementation in *madmom v0.17.dev0* [29]. It consists of an LSTM-based model trained in ten datasets spanning Western genres, and Carnatic, Cretan and Turkish music excerpts. We also include the beat tracker from Ellis [32], which estimates a global tempo and then uses dynamic programming to find the best set of beats that reflect such tempo. As a final baseline, we include the TCN of Section 2.3 trained with the Western datasets from Section 2.1 (TCN-BL).

2.6 Evaluation metrics

We use as our main metric F-measure [33], along with the continuity-based metrics [34, 35] CMLt (“correct metrical level”) which corresponds to the ratio between correct and annotated beats, and AMLt (“allowed metrical level”) which accepts phase errors of half a beat period or octave errors in estimation. For the computational cost of the models, we simply report the time they take to train by using in-build timing functions in the code.

3. EXPERIMENTS AND RESULTS

3.1 Performance of models

Figure 1 shows the F-measure results for the TCN models trained for *candombe* and for *samba* with different amounts of data using each of the training strategies, as well as BayesBeat, computed as the bootstrapped results of ten experiments (95% confidence) with different random seeds for each combination of model and data amount.

A first striking observation is that for both beats and downbeats, the performance curve for most models has a small positive slope, which means it is indeed possible to nearly achieve best model performance (which would require training with full dataset) by just training with few samples. This is particularly true for the estimation of beat, for which models rapidly reach F-measure scores above 80% with less than a minute of data in both *candombe* and *samba* for almost all configurations. This is an interesting

Model	*Candombe*						*Samba*					
	Beat			Downbeat			Beat			Downbeat		
	CMLt	AMLt	F	CMLt	AMLt	F	CMLt	AMLt	F	CMLt	AMLt	F
BayesBeat_0.67	95.0	95.0	96.1	82.2	92.0	81.7	70.5	74.2	78.8	57.4	72.9	57.9
BayesBeat_12.33	99.6	99.6	99.6	99.8	99.8	99.4	93.5	96.0	96.7	92.5	94.9	92.9
BayesBeat_all	98.6	98.6	98.9	98.8	98.8	98.4	94.0	96.0	96.9	92.0	95.3	92.5
TCN-FSA_0.67	96.1	96.2	96.4	87.7	90.9	87.8	83.4	86.3	87.7	60.7	80.7	73.2
TCN-FSA_12.33	98.1	98.1	98.1	93.9	96.3	93.4	94.3	95.7	96.9	91.1	95.0	93.8
TCN-FSA_all	99.0	99.3	99.1	99.2	99.6	98.6	95.7	98.2	98.1	95.7	98.0	96.4
TCN-FTA_0.67	99.2	99.2	99.1	93.5	96.8	92.8	92.9	95.7	96.2	69.4	90.0	82.3
TCN-FTA_12.33	99.6	99.6	99.4	96.3	99.7	95.5	95.8	97.2	98.0	92.5	96.9	95.6
TCN-FTA_all	99.5	99.5	99.3	99.3	99.8	98.8	97.3	98.7	98.7	96.7	97.2	97.5
TCN-BL	11.1	18.7	15.9	14.9	31.9	4.1	46.5	65.6	60.0	5.9	52.5	9.6
Ellis [32]	34.8	38.1	38.0	-	-	-	82.3	87.6	87.1	-	-	-
Böck [4]	11.7	14.4	11.5	26.7	40.3	0.5	46.9	76.0	66.4	5.2	66.6	2.0

Table 1: Mean F-measure (F) and continuity scores (CMLt, AMLt) in beat and downbeat tracking tasks across both genres.

result, meaning that not much gain in performance is expected with the increase of annotations for such datasets. An end-user could annotate less that a minute of data and yet obtain decent performances. The same holds for downbeat in *candombe*, but not in *samba*. In the latter, there is a clear gain in adding more data, which has to do with the differences between the two rhythms, as discussed below.

Differences between *candombe* and *samba*. Observing the results in Figure 1, we see that the models tend to require more data to achieve better performance on *samba* than on *candombe*, and the uncertainty about the performance for *samba* is larger. Our intuition behind this result is that, as mentioned in Section 1.2, because *samba* has a bigger combination of timbres and pitches than candombe, the decision of what snippets to annotate (i.e., the sampling) might be more critical for the former than for the latter, e.g., ensuring timbre representation.

Best model configuration. The best performing configuration for beat and downbeat tracking in both music traditions is the fine-tuned TCN model with data augmentation (FTA). Particularly, data augmentation produced significant improvement in performance for downbeat tracking in samba. Interestingly, for the adaptive setting concerned in this work, the BayesBeat baseline is competitive with the TCN model, especially considering the computational cost (see Section 3.2).

Comparison with off-the-shelf benchmarks. Table 1 shows the performance of the TCN and the BayesBeat baseline for different data subsets, namely the smallest and largest subsets, and the full dataset. It also shows the performance of the three off-the-shelf baselines explained in Section 2.5. In alignment with previous works [12, 21], the models trained with Western music (TCN-BL and Böck [4]) perform very poorly in *candombe*, and reach only about 66% F-measure in *samba*, both significantly lower than the performance of the same models in Western music genres. The model of Ellis [32] scores considerably better, but is not consistent in both datasets. This shows the necessity of adapting meter tracking models to these music genres, as even the models trained with the smallest subsets of data (0.67 min) outperform the baselines.

3.2 How much time do the models take to train?

Our analysis is motivated by the adaptation of meter tracking models in real-world use cases. For this adaptation to make sense it has to be done quickly. In this regard, we estimate the time each model configuration takes in training, and contrast it with the BayesBeat baseline. Figure 2 shows how the train duration varies with the size of the train set for *samba* (very similar results were obtained for *candombe*). The TCN takes about the same time in both *samba* and *candombe*, with a minimum of about 100 s for the smallest subset. Among the TCN configurations, the most expensive ones use data augmentation. This makes sense given that more data is used for training. As expected, the BayesBeat trains significantly faster than the TCN, taking on average 1.62 s to train with 0.67 min of data, and being in the order of 50 to 150 times faster than the TCN when data augmentation is not used. This big gap in computing time, together with the results of Figure 1 and Table 1, makes BayesBeat an overall good alternative for adapting meter tracking to these Latin American music. We observed that all configurations take about the same inference time, around 25 s for the full test set.

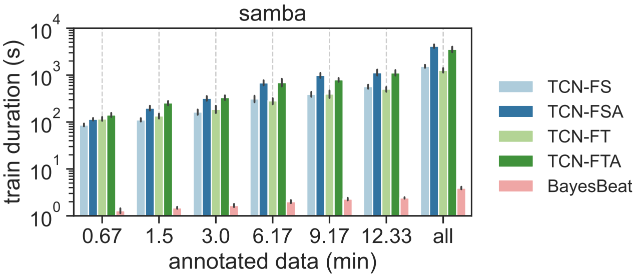

Figure 2: Training time for the different amounts of data.

3.3 When can we train with small data?

Our intuition is that the more variability in the data (in terms of meters, rhythmic patterns and instrumentation), the harder it is for a model to learn with small data. This aligns with our experiments in the adaptability of these methods to *samba* and *candombe*, and also agrees with the musicological insights of Section 1.2. To have a more quantitative understanding of this, we derived a bar profile for each type of music. First we extract a feature map from each excerpt using the beat/downbeat annotations to time-quantize a locally normalized onset strength function [36] at the tatum scale — this was done with the *carat* [37] toolbox, considering the tatum duration as one quarter of the time-span between successive beats. Then, for each dataset, we summarize these feature maps across time, which results in a distribution of feature values per tatum. To allow an analysis of these profiles in different regions of the spectrum, we compute the onset strength in two frequency bands (20 Hz to 200 Hz; and > 200 Hz). We present these distributions as violin plots in Figure 3 for *candombe*, *samba*, and for the Ballroom dataset.

In Figure 3, we verify that for some tatums strength distributions are concentrated around 1 or 0, indicating a strong characteristic accent or lack thereof at that point of the bar respectively. High variance, in its turn, means "fuzzyness" in the rhythm pattern, which could justify the difficulty in learning that rhythm, specially with small data.

Samba, which has eight tatums per bar (2/4 meter), is known for having a strong metrical accent at beat 2, which we may readily identify in its low-frequency channel at tatum 5. The first beat also has a high median value but is less "deterministic" due to its high variance. In turn, the low-frequency profile of *candombe* displays a high-variance downbeat, no accent on beat 2, and strong accents on beats 3 and 4, but also a strong contrametric accent at tatum 4. These characteristics could help explain why the off-the-shelf beat tracking models, which expect beats to be accented, perform worst on *candombe*. Looking back at *samba*, we see that tatums 2 and 3 show small standard deviations and correspond to "off" tatums; together with beat 2, they make three out of eight tatums that exhibit very small variance in the low channel. In *candombe*, besides tatum 4, tatums 2, 3, 7, 8, 9, 14, and 16 also present small variance. This abundance of "anchor" points could justify why adaptation in *candombe* came with little data.

In Ballroom, we clearly see that beats are distinct for having high strength and low variance in both channels, whereas the rest of the tatums show no clear trend. Its few reference points could pose a challenge for learning models. Furthermore, beat patterns (the combination the four tatums in-between beats, including the beat itself) are also indistinguishable from one another, which could aggravate this matter. To test these observations, we trained a set of models from scratch for Ballroom using the same methodology that for *samba* and *candombe*. Results are depicted in Figure 4. The performance results correlate with the intuition that Ballroom is a more challenging dataset given that it comprises multiple genres, and also that for learning beat and downbeat more data would be needed.

4. CONCLUSIONS AND FUTURE WORK

We adapted a meter tracking model using small quantities of data to work in particular Latin American music traditions, namely *samba* and *candombe*. We showed that, under certain homogeneity conditions, it is indeed possible to train such models with a few minutes of annotated data and training cycles, and obtain almost full-dataset performance. This result has promising consequences in real-world applications, as it opens the possibility of adapting such models to other music genres with modest labeling efforts. The most competitive model is a fine-tuned TCN with data augmentation, whereas BayesBeat is a good option under computational cost constraints. In the future, we will investigate rhythm complexity metrics that could serve to predict the amount of annotated data needed to adapt meter tracking models to particular music genres.

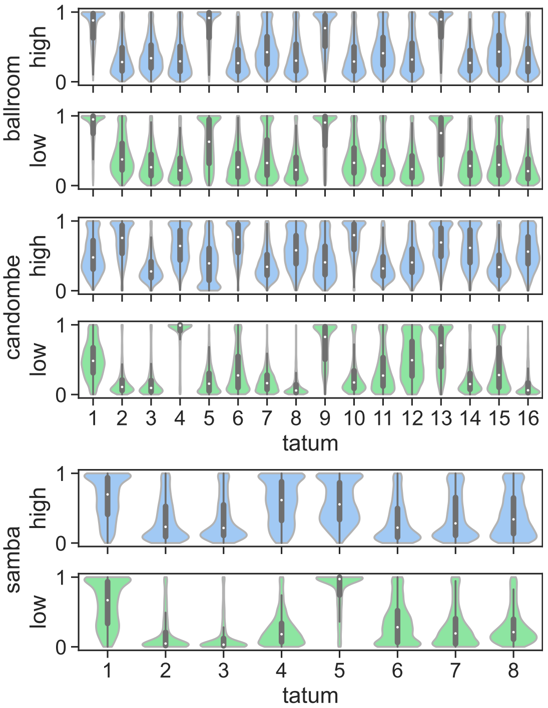

Figure 3: Tatum strength distribution per frequency band for Ballroom (just 4/4 tracks), *candombe*, and *samba*.

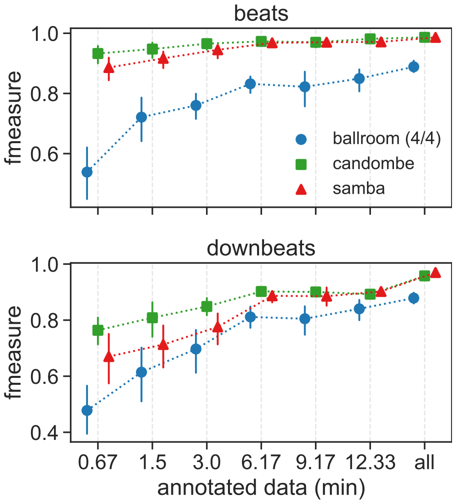

Figure 4: TCN-FS performance in Ballroom.

5. ACKNOWLEDGEMENTS

This study was partially funded by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) — Finance Code 001.

6. REFERENCES

[1] A. Klapuri, A. Eronen, and J. Astola, “Analysis of the meter of acoustic musical signals,” *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 1, pp. 342–355, 2006.

[2] B.-Y. Chen, W.-H. Hsu, W.-H. Liao, M. A. M. Ramírez, Y. Mitsufuji, and Y.-H. Yang, “Automatic DJ transitions with differentiable audio effects and generative adversarial networks,” in *Proc. 2022 IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2022, pp. 466–470.

[3] A. Srinivasamurthy, A. Holzapfel, K. K. Ganguli, and X. Serra, “Aspects of tempo and rhythmic elaboration in Hindustani music: A corpus study,” *Frontiers in Digital Humanities*, vol. 4, 2017. [Online]. Available: https://www.frontiersin.org/article/10.3389/fdigh.2017.00020

[4] S. Böck, F. Krebs, and G. Widmer, “Joint beat and downbeat tracking with recurrent neural networks,” in *Proc. 17th Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, New York, USA, Aug. 2016, pp. 255–261.

[5] S. Durand, J. P. Bello, B. David, and G. Richard, “Robust downbeat tracking using an ensemble of convolutional networks,” *IEEE Trans. Audio, Speech, Language Process.*, vol. 25, no. 1, pp. 76–89, Jan 2017.

[6] M. Fuentes, B. McFee, H. Crayencour, S. Essid, and J. Bello, “Analysis of common design choices in deep learning systems for downbeat tracking,” in *Proc. 19th Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Paris, France, Sep. 2018, pp. 106–112.

[7] M. Heydari, F. Cwitkowitz, and Z. Duan, “Beatnet: CRNN and particle filtering for online joint beat downbeat and meter tracking,” in *Proc. 22nd Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Online, Nov. 2021, pp. 270–277.

[8] M. E. P. Davies, S. Böck, and M. Fuentes, *Tempo, Beat and Downbeat Estimation*. https://tempobeatdownbeat.github.io/tutorial/intro.html, Nov. 2021. [Online]. Available: https://tempobeatdownbeat.github.io/tutorial/intro.html

[9] C. N. Silla Jr., A. L. Koerich, and C. A. A. Kaestner, “The Latin music database,” in *9th Int. Conf. Music Inf. Retrieval (ISMIR)*, Philadelphia, USA, Sep. 2008, pp. 451–456.

[10] E. Cano, F. Mora-Ángel, G. A. López Gil, J. R. Zapata, A. Escamilla, J. F. Alzate, and M. Betancur, “Sesquialtera in the Colombian bambuco: Perception and estimation of beat and meter,” in *Proc. 21st Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Montreal, Canada, Oct. 2020, pp. 409–415.

[11] G. M. Sarria M., J. Diaz, and C. Arce-Lopera, “Analyzing and extending the Salsa music dataset,” in *Proc. XXII Symp. Image, Signal Process., Artif. Vision (STSIVA)*, Bucaramanga, Colombia, Apr. 2019, pp. 1–5.

[12] L. Nunes, M. Rocamora, L. Jure, and L. W. P. Biscainho, “Beat and downbeat tracking based on rhythmic patterns applied to the Uruguayan Candombe drumming,” in *Proc. 16th Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Málaga, Spain, Oct. 2015, pp. 246–270.

[13] D. Fiocchi, M. Buccoli, M. Zanoni, F. Antonacci, and A. Sarti, “Beat tracking using recurrent neural network: a transfer learning approach,” in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, Rome, Italy, Sep. 2018, pp. 1929–1933.

[14] A. S. Pinto, S. Böck, J. S. Cardoso, and M. E. P. Davies, “User-driven fine-tuning for beat tracking,” *Electronics*, vol. 10, no. 13, Jun. 2021.

[15] K. Yamamoto, “Human-in-the-loop adaptation for interactive musical beat tracking,” in *Proc. 22nd Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Online, Nov. 2021, pp. 794–801.

[16] S. Böck and M. E. P. Davies, “Deconstruct, analyse, reconstruct: How to improve tempo, beat, and downbeat estimation,” in *Proc. 21st Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Montreal, Canada, Oct. 2020, pp. 574–582.

[17] F. Krebs, S. Böck, and G. Widmer, “Rhythmic pattern modelling for beat and downbeat tracking from musical audio,” in *Proc. 14th Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Curitiba, Brazil, Nov. 2013, pp. 227–232.

[18] J. Fonseca, M. Fuentes, F. Bonini Baraldi, and M. E. P. Davies, “On the use of automatic onset detection for the analysis of Maracatu de baque solto,” in *Perspectives on Music, Sound and Musicology: Research, Education and Practice*, L. C. Castilho, R. Dias, and J. F. Pinho, Eds. Cham, Switzerland: Springer, 2021, pp. 209–225.

[19] G. Gonçalves and O. Costa, *The Carioca Groove: The Rio de Janeiro’s Samba Schools Drum Sections*. Rio de Janeiro, Brazil: Groove, 2000.

[20] M. Rocamora, L. Jure, B. Marenco, M. Fuentes, F. Lanzaro, and A. Gómez, “An audio-visual database of Candombe performances for computational musicological studies,” in *Proc. II Congreso Int. de Ciencia y Tecnología Musical (CICTeM)*, Buenos Aires, Argentina, Sep. 2015, pp. 17–24.

[21] L. S. Maia, P. D. Tomaz Jr., M. Fuentes, M. Rocamora, L. W. P. Biscainho, M. V. M. Costa, and S. Cohen, “A novel dataset of Brazilian rhythmic instruments and some experiments in computational rhythm analysis,” in *Proc. 2018 AES Lat. Am. Congr. Audio Eng. (AES LAC)*, Montevideo, Uruguay, Sep. 2018, pp. 53–60.

[22] F. Gouyon, A. Klapuri, S. Dixon, M. Alonso, G. Tzanetakis, C. Uhle, and P. Cano, "An experimental comparison of audio tempo induction algorithms," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 5, pp. 1832–1844, Sep. 2006.

[23] M. E. P. Davies, N. Degara, and M. D. Plumbley, "Evaluation metrics for musical audio beat tracking algorithms," Queen Mary University of London, London, UK, Tech. Report, Oct. 2009.

[24] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Trans. Speech, Audio Process.*, vol. 10, no. 5, pp. 293–302, Jul. 2002.

[25] U. Marchand and G. Peeters, "Swing ratio estimation," in *Proc. 18th Int. Conf. Digital Audio Effects (DAFx)*, Trondheim, Norway, Dec. 2015, pp. 423–428.

[26] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC music database: Popular, Classical, and Jazz music databases," in *Proc. 3rd Int. Conf. Music Inf. Retrieval (ISMIR)*, Paris, France, Oct. 2002, pp. 287–288.

[27] M. Goto, "Development of the RWC music database," in *Proc. 18th Int. Congr. Acoust. (ICA)*, Kyoto, Japan, Apr. 2004, pp. I–553–556.

[28] R. M. Bittner, M. Fuentes, D. Rubinstein, A. Jansson, K. Choi, and T. Kell, "mirdata: Software for reproducible usage of datasets," in *Proc. 20th Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, 2019.

[29] S. Böck, F. Korzeniowski, J. Schlüter, F. Krebs, and G. Widmer, "madmom: a new Python Audio and Music Signal Processing Library," in *Proc. 24th ACM Int. Conf. Multimedia*, Amsterdam, The Netherlands, Oct. 2016, pp. 1174–1178.

[30] N. Whiteley, A. Cemgil, and S. Godsill, "Bayesian modelling of temporal structure in musical audio," in *Proc. 7th Int. Conf. Music Inf. Retrieval (ISMIR)*, Victoria, Canada, Oct. 2006, pp. 29–34.

[31] A. Holzapfel, F. Krebs, and A. Srinivasamurthy, "Tracking the "odd": Meter inference in a culturally diverse music corpus," in *Proc. 15th Int. Soc. Music Inf. Retrieval Conf. (ISMIR)*, Taipei, Taiwan, Oct. 2014, pp. 425–430.

[32] D. P. W. Ellis, "Beat tracking by dynamic programming," *Journal of New Music Research*, vol. 36, no. 1, pp. 51–60, Mar. 2007.

[33] S. Dixon, "Evaluation of the audio beat tracking system BeatRoot," *J. New Music Res.*, vol. 36, no. 1, pp. 39–50, 2007.

[34] S. W. Hainsworth, "Techniques for the automated analysis of musical audio," Ph.D. dissertation, Department of Engineering, Cambridge University, 2003.

[35] A. P. Klapuri, A. J. Eronen, and J. T. Astola, "Analysis of the meter of acoustic musical signals," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, no. 1, pp. 342–355, Jan. 2006.

[36] M. Rocamora, L. Jure, and L. W. P. Biscainho, "Tools for detection and classification of piano drum patterns from Candombe recordings," in *Proc. 9th Conf. Interdisciplinary Musicology (CIM14)*, Berlin, Germany, Dec. 2014, pp. 382–387.

[37] M. Rocamora and L. Jure, "carat: Computer-Aided Rhythmic Analysis Toolbox." in *Proc. Analytical Approaches World Music (AAWM)*, Birmingham, UK, Jul. 2019.