ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9414601

ON THE PREPARATION AND VALIDATION OF A LARGE-SCALE DATASET OF SINGING TRANSCRIPTION

Jun-You Wang, Jyh-Shing Roger Jang

Dept. CSIE, National Taiwan University, Taiwan

ABSTRACT

This paper proposes a large-scale dataset for singing transcription, along with some methods for fine-tuning and validating its contents. The dataset is named MIR-ST500, which consists of more than 160,000 notes from 500 pop songs. To create this large-scale dataset, we set some labeling criteria and ask non-experts to label notes. We also perform some adjustments on the annotation to correct minor errors. Finally, to validate the dataset, we train a singing transcription model on MIR-ST500 dataset and evaluate it on various datasets. The result shows that we can certainly construct a better singing transcription model for various purposes using MIR-ST500, which is properly labeled and validated.

Index Terms— Automatic singing transcription, dataset preparation, dataset validation, music information retrieval

1. INTRODUCTION

At the era of machine learning, a reliable large-scale dataset is of utmost importance for successful applications. However, for some tasks in MIR (music information retrieval), the creation of large-scale datasets is difficult due to the requirement of accurate labels. One example is automatic singing transcription (AST) which focuses on converting singing voice to sheet music. Such a dataset for AST is hard to come by due to the tremendous amount of effort needed to label the onsets/offsets/pitch of each note precisely. One of the evaluation frameworks [1] set the "onset tolerance" to 50ms, which, in terms, requires a strict labeling criterion, making the labeling process extremely time-consuming and leading to relatively small scale of the datasets [1-2].

In this work, we take another route by setting several easy rules for non-expert transcribers to label a large number of notes. Then we perform several automatic adjustments on the onsets to eliminate most errors. Admittedly, there may still be some undetected errors. But the advantage of the large-scale dataset usually outweighs minor errors in the dataset, as revealed by our task of dataset validation in Section 4.

2. RELATED WORK

There are several datasets that could be used for AST. We classify these datasets into two categories: "created for AST" and "not created for AST".

Created for AST: ISMIR2014 dataset [1] and TONAS dataset [2] are created for AST. The scale of these datasets is not large, and these datasets are all labeled by experts.

Not created for AST: AIST annotation of "RWC Music Database: Popular Music" [3] provides note-level annotations of 100 pop songs, which could be used for AST. However, the annotation is not created specifically for AST, so we do not know if the annotation is accurate enough. DALI dataset [4] is another dataset with note annotation of singing voice. However, DALI's transcription is created automatically with inevitable errors [4].

For the adjustment of onset labels in our dataset, we use MIR-1k dataset [5] to train a model to perform onset adjustment automatically, as will be explained in Section 3.4.

For the AST model, traditional models for AST employ Hidden Markov Model (HMM) to model the transition of notes [6-8]. Recently, Long Short-Term Memory (LSTM) model [9] and ResNet-18 [10] are also utilized due to their end-to-end nature.

3. METHODOLOGY

3.1. Dataset Overview

The proposed "MIR-ST500" dataset consists of 500 pop songs, mostly in Chinese, from Youtube. The vocal parts in the songs are monophonic, but they may contain instruments. The duration of the dataset is about 30 hours, and the number of notes is 162,438. The dataset is available at http://mirlab.org/dataset/public now.

3.2. Annotation Procedure

We assigned non-experts to one of two groups, "transcribers" and "verifiers". To start with, a transcriber downloads a pop song from Youtube and labels its notes one by one. To make the process more efficient, a baseline AST program can be used to generate the initial labeling. Transcribers are encouraged to use a MIDI editor to label notes, so they can listen to their annotation at left channel and the original music in the right channel simultaneously, making comparison/labeling/correction easier.

After labeling, a verifier is asked to verify and decide if the annotation is acceptable. If the verifier finds any mistake, the transcriber should relabel the song until no mistake is found by the verifier. Only those labels agreed by both transcriber and verifier are included in the dataset.

To make the annotation more objective and consistent, we have defined several intuitive labeling rules shown next.

- **Pitch**: The groundtruth is labeled as the pitch that should be sung on, i.e., the "score pitch", which should be integers of semitones (MIDI numbers).
- **Onset**: Onsets should be placed before the voiced part of the note. If there is an unvoiced part at the beginning, transcribers can place onsets at any place within the leading unvoiced segment. For example, if the lyrics of a note is

"sing", the onset can be placed at either the beginning of "s" or the beginning of "i". Since it is hard to identify the start position precisely, we gave labelers more flexibility in order to make the labeling process easier.

- **Offset**: For the offset of notes, we did not give transcribers clear guide on how to place offset. Transcribers were only asked to place offset at the time they think the voice ends. If the voice does not end clearly, the offset should be placed at the onset of next note.

Beside these rules, we also provided a "baseline AST program" to accelerate the annotation process. This program first uses Spleeter [11] to obtain vocal part of a song, and then uses the CNN onset detector proposed in [12] to determine onsets, and use the pitch tracking algorithm proposed in [13] to obtain pitch of each frame. The final "score pitch" is the rounded average of frames' pitch within that note. The offset is determined using an energy-based voiced detection algorithm implemented by [13].

3.3. Concerns About the Dataset

With the above annotation procedure, 162,438 notes from 500 songs are created. However, several concerns still exist, which are listed below together with our methods to reduce them.

- **Score pitch**: The first concern is that transcribers do not have formal music training, which might have led to some transcription error on score pitch. This was alleviated by letting transcribers to freely select songs he/she is comfortable with. Moreover, by playback the song and the labeled notes simultaneously, most people can readily identify the pitch error (if any).
- **Initial label bias:** The second concern is that we provided a baseline AST program for transcribers, which might have led to some bias in favor of the baseline AST program. To alleviate this problem, we performed an extra post-processing step to adjust onsets automatically, as described in Section 3.4. This could reduce some of the initial label bias.
- **Onset criterion:** The third concern is that we did not define the rule of labeling onset strictly, which may cause some ambiguity. In particular, if an onset is labeled at the start of an unvoiced segment, but an AST program predicts it at the start of voiced segment, the prediction may be considered wrong if the deviation exceeds the given tolerance. To make sure this is not a big concern, we conducted two experiments. First, we computed the average duration of unvoiced segment in MIR-1k dataset [5], a dataset that has similar properties with MIR-ST500, in Section 3.5. And then we trained an AST model using MIR-ST500 dataset in Section 3.6, and then evaluated its performance in Section 4. The results provide some indirect evidence, showing that although the onset annotation of MIR-ST500 dataset may be a little bit inconsistent, it is still acceptable for AST modeling.

3.4. Post-processing on Onset Labels

After the labeling process, we performed an extra post-processing step on onsets. We focus especially on those notes that have unvoiced phoneme at the beginning, since onsets of these notes are much difficult to place. We have defined criterion to label these onsets, i.e., they should be placed within the leading unvoiced segment. In this step, we adjusted these onsets to force them to follow the criterion. We first trained an "unvoiced frame classifier" to identify unvoiced segments in MIR-ST500 dataset, and then moved onsets to their nearest unvoiced segment (if any).

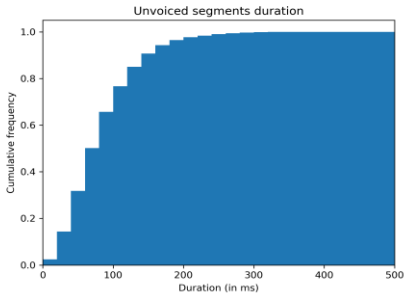

Figure 1. The cumulative distribution of unvoiced segments' duration in MIR-1k dataset.

We used MIR-1k dataset [5] to train the classifier. MIR-1k dataset consists of 1000 song clips (about 133min) in Chinese. It provides the label of "unvoiced frame", so we can train our model on it. Since most songs in MIR-ST500 are also sang in Chinese, the gap between training data and test data is relatively small.

The model is a neural network that takes Mel-Frequency Cepstral Coefficients (MFCC), MFCC delta and delta-delta as input. The model itself consists of 4 linear layers with ReLU activation function. A sigmoid function is applied to determine the final unvoiced probability. After training the model, we predicted each frame in MIR-ST500 by using Viterbi algorithm [14], with transition probability also obtained from MIR-1k.

Finally, we computed the shortest time difference between the onset and any unvoiced segment (predicted by the model). If for an onset label, the time difference is not zero and is smaller than a threshold, then it is considered as with "small error", and will be moved to the boundary of its nearest unvoiced segment.

The reason to set a threshold is simply because we do not want to adjust those onsets of notes that do not have a leading unvoiced phoneme. These onsets are usually far away from any unvoiced segments, so we can set a threshold to avoid moving them erroneously. Note that such a threshold should be large enough to cover most of the small errors committed by transcribers, but also small enough to avoid erroneous move. Here, we set the threshold to 0.05 second, or 50ms, empirically.

3.5. Duration of Unvoiced Segments

In this section, we calculate the distribution of unvoiced segments duration in MIR-1k dataset. This helps us to find out the level of ambiguity introduced by the onset labeling criteria.

In total, there are 5927 unvoiced segments in MIR-1k, and the average duration of them is 99.8ms (4.99 frames). The cumulative distribution is shown in Figure 1. About 70% unvoiced segments have duration shorter than 100ms. This indicates that even if we give transcribers more tolerance, the "ambiguity" does not exceed 100ms in most unvoiced segments. However, we still do not know if the onset accuracy of the dataset is acceptable when onset tolerance is set to ±50ms (which is frequently adopted in previous works [1, 8, 10, 15]). This should be verified by experiments in Section 4.

3.6. Singing Transcription Model Trained on MIR-ST500

Dataset	Size	Year	Vocal only	Audio	Method	Pitch criterion
RWC (popular)	100 songs[1] >30000 notes	2002 (music) 2006 (label)	No	Pop songs Copyright-cleared	Not clear	Score pitch
TONAS	72 songs 2983 notes	2013	Yes	Flamenco songs	Labeled by experts	Floating-point note pitch
ISMIR2014	38 songs 2153 notes	2014	Yes	Pop songs and children songs	Labeled by experts	Floating-point note pitch
DALI	5358 songs[2] >1.6M notes	2018	No	Pop songs From Youtube	Automatic Alignment	Score pitch (with many errors)
MIR-ST500	500 songs >160000 notes	2021	No	Pop songs From Youtube	Labeled by non-experts	Score pitch

Table 1. The comparison between MIR-ST500 and previous datasets.

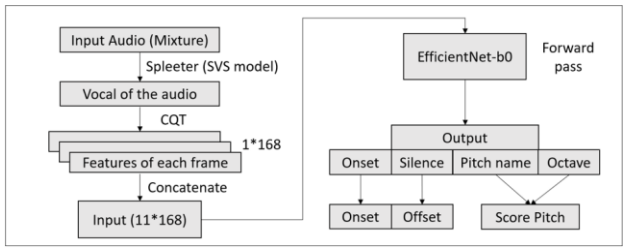

Figure 2. The pipeline of our sample AST model.

In this section, we describe a pipeline of using MIR-ST500 to train an AST model. We will then evaluate this model on various datasets (in Section 4.3 and 4.4), in order to demonstrate the feasibility of the dataset. The performance of our model can also serve as the baseline of MIR-ST500 dataset.

Figure 2 shows the pipeline of our model. Our AST program is based on EfficientNet-b0 [16], which is a CNN model that reached state-of-the-art performance on ImageNet-1k [17], while the model size is significantly lower than other models with similar performance. To prepare the training data, we first resample the audio to 44100Hz (if needed), and then use Spleeter [11] to extract vocal part. Then we compute constant-Q transform (CQT) of the audio with a hop length of 1024 sample points (about 23ms) and 24 bins per octave (from C2 to C9) to obtain a 168-dimensional vector for each frame. The network input is the concatenation of the vectors from ±5 frames (11 frames in total, corresponding to the context of about 255ms), which can be viewed as a 1-channel "image" of 168x11 resolution.

The output of the network is an 18-dimensional vector. The first two outputs are the "onset probability" and "silence probability". The 3rd to 14th outputs are the probability of each "pitch name", from C to B, while the remaining 4 outputs are the probability of each "octave", from 2 to 5. By applying onset threshold and silence threshold, we can segment audio into notes, each contains several frames. The most commonly appeared pitch is considered as the score pitch of a note. The onset threshold is chosen to maximizes COn f1-score of the training set. The silence threshold is set to 0.5 empirically.

4. EXPERIMENTS

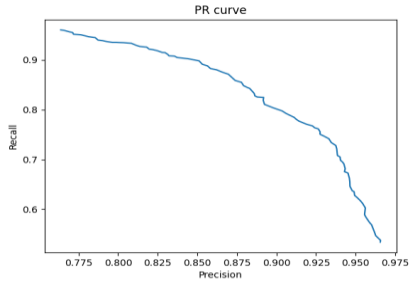

Figure 3. The precision-recall curve of our unvoiced frame detector on MIR-1k test set.

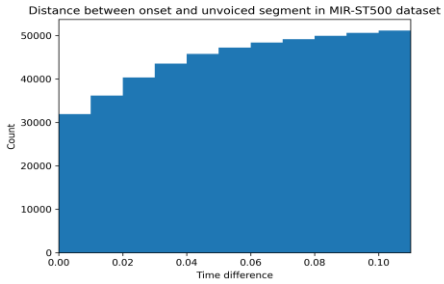

Figure 4. The cumulative count of "the distance in time between onset and the closest unvoiced segment" on MIR-ST500 dataset.

The comparison between MIR-ST500 dataset and others, including ISMIR2014 dataset [1], TONAS [2], "RWC database: Popular music" [3, 18] and DALI dataset [4], is shown in table 1.

4.1. Results of Unvoiced Frame Classifier

We split MIR-1k into a 900-clip training set and a 100-clip test set, and then trained our model for 50K steps with a batch size of 50. Figure 3 shows the precision-recall curve on the test set, which is obtained by adding the unvoiced log probability of each frame by the same amount of prior p. We change the value of p to make the precision equals to recall (the resulted R-precision is 86.96%). This model (and the prior p) is then used to test MIR-ST500 to see if onsets are placed at unvoiced segments properly.

By using this model, we identified 61,032 unvoiced segments, while the dataset itself contains 162,438 labeled onsets. Figure 4 shows the cumulative count of "the distance between the onset

[1] This dataset contains 100 songs, but 6 of them (No. 3, 5, 8, 10, 23, and 66) have multiple singers, so we only use the remaining 94 songs to evaluate our model.

[2] This dataset originally contains 5358 songs, but we can only download 4726 of them from Youtube using the provided script (in July 2020), so we only use these 4726 songs to evaluate our model.

Dataset	ISMIR2014	MIR-ST500 (test)	MIR-ST500 (train)
COnPOff	49.55%	45.78%	**60.00%**
COnP	63.63%	66.63%	**74.39%**
COn	**79.16%**	75.44%	78.76%
Dataset	**TONAS**	**RWC**	**DALI**
COnPOff	9.57%	6.11%	3.61%
COnP	19.65%	10.32%	9.80%
COn	42.41%	69.29%	44.35%

Table 2. F1-score on various datasets with 50ms onset tolerance.

Dataset	DALI	MIR-ST500 (test)	RWC	ISMIR2014
COn(50ms)	44.35%	75.44%	69.29%	79.16%
COn(100ms)	66.87%	84.82%	80.01%	92.57%

Table 3. F1-score on datasets with different onset tolerance.

and the closest unvoiced segment". 31,911 of the labeled onsets are placed within an unvoiced segment. 15,340 of them satisfy the heuristics discussed in Section 3.4, namely, "the distance is less than 50ms". These 15,340 onsets were then adjusted automatically, which account for 9.4% of the total onsets. In comparison with the total number of unvoiced segments (61,032), this may seem quite high (25.1%). However, such a mistake (most of them are within 20ms, according to figure 4) could also be committed by the unvoiced detector itself, so we think this is still acceptable.

4.2. Evaluation Metrics of Singing Transcription

We use the evaluation metrics proposed in [1], including COn (Correct Onset), COnP (Correct Onset and Pitch) and COnPOff (Correct Onset, Pitch and Offset), to evaluate our AST model. If the onset/pitch/offset difference between two notes are within a certain threshold, the onset/pitch/offset is "correct". By comparing groundtruth and transcribed notes, we can find out the number of notes that satisfy COn, COnP and COnPOff, and further compute f1-score of each metric.

In our experiments, the pitch threshold is set to 50 cents, while the offset threshold is set to max(50ms, 0.2*note duration). These thresholds are used frequently in previous papers [8, 10, 15].

4.3. Results of the Singing Transcription Model

To build the sample AST model, we split MIR-ST500 into a 400-song training set and a 100-song test set. After training for 300K steps with a batch size of 50, the performance on various datasets are showed in table 2. Several findings are listed next.

Pitch: Our model scores high COnP f1-score on MIR-ST500 test set and ISMIR2014 dataset, but does not score high on RWC and DALI. By examining RWC and DALI, we found that the pitch annotations of RWC and DALI are not good enough since some of the pitch annotations are one or two octaves higher than the vocal part. Also, our model scores higher on MIR-ST500 test set than ISMIR2014, which may be due to the fact that the pitch labeling criterion of ISMIR2014 dataset is different from MIR-ST500, as we described in Table 1.

Onset: For COn, our model performs well on MIR-ST500 test set, ISMIR2014, and RWC-MDB-P, but not so well on DALI and TONAS. For TONAS, this is due to the low recall rate, since it has 2983 notes in total, but the model only outputs 1386. The reason may be due to the annotating criteria of the dataset itself. The labelers of TONAS tend to split notes when there is vibrato or portamento. For DALI, both precision and recall are low, which is likely due to inaccuracy of labeling.

Method\Metrics	COnPOff	COnP	COn
EfficientNet-b0	45.78%	66.63%	75.44%
Baseline	24.81%	39.24%	61.18%

Table 4. Comparison of f1-scores on MIR-ST500 test set.

Method\Metrics	COnPOff	COnP	COn
Tony [15]	50%	**68%**	73%
Fu & Su [10]	**59.4%**	NA	78.6%
EfficientNet-b0	49.55%	63.63%	**79.16%**

Table 5. Comparison of f1-score with previous approaches on ISMIR2014 dataset. The "NA" cell means this metric is not reported in the paper.

Onset tolerance: Table 3 shows the COn f1-score with onset tolerance of 50ms and 100ms of our model. The COn difference between DALI and MIR-ST500 test set is 31% when onset tolerance is set to 50ms, but if onset tolerance is set to 100ms, the difference shrinks to 18%. This may indicate that the onset annotations of DALI have larger "error range". In contrast, we can also conclude that the onset error range of MIR-ST500 is lower, since the difference between 50ms (75.44%) and 100ms (84.82%) is only 9.38%, which means the ±50ms range is enough to cover most of the onsets.

4.4. Model Comparison

In this section, we compare our (EfficientNet-b0) model with some other models, including the "baseline" model used to help transcribers label MIR-ST500. We set the onset tolerance to 50ms here. Table 4 shows that EfficientNet-b0 model does outperform the baseline model on MIR-ST500 test set. This indicates that MIR-ST500 does not give too much advantage to the baseline AST program, since a model trained on MIR-ST500 training set can easily outperform baseline model by at least 14% on every metric. Table 5 shows that our model trained on MIR-ST500 is competitive with other state-of-the-art models, despite the different pitch labeling criterion between training set (MIR-ST500) and test set (ISMIR2014), which gives disadvantage to our model. This also proves that although there are some concerns about MIR-ST500 (as mentioned in Section 3.3), it is still a useful large-scale dataset that can be utilized to create a feasible AST model.

5. CONCLUSIONS

This paper proposes a large-scale dataset, MIR-ST500, and performs several automatic adjustments to reduce the error of labels. We also trained an AST model on MIR-ST500 dataset, and evaluated its performance on other datasets to demonstrate the feasibility of the dataset. The result also shows that with the help of a baseline program, verification process, post-processing and the use of converting annotation to MIDI, even non-experts can still create useful MIR-related datasets.

To further improve the reliability, as a future work, the validation of the test set (or a subset of this 100-recording test set) by experts can be conducted to make it more accurate.

6. REFERENCES

[1] E. Molina, A. M. Barbancho-Perez, L. J. Tardón, I. Barbancho-Perez, "Evaluation framework for automatic singing transcription," in *Proceedings of the 15th International Society for Music Information Retrieval Conference (ISMIR 2014)*, pp.567-572, October 2014.

[2] E. Gómez and J. Bonada, "Towards computer-assisted flamenco transcription: An experimental comparison of automatic transcription algorithms as applied to a cappella singing," *Computer Music Journal*, 37(2):73–90, 2013.

[3] M. Goto, "AIST Annotation for the RWC Music Database," in *Proceedings of the 7th International Conference for Music Information Retrieval (ISMIR 2006)*, pp.359-360, 2006.

[4] G. Meseguer-Brocal, A. Cohen-Hadria, G. Peeters, "DALI: a large Dataset of synchronized Audio, LyrIcs and notes, automatically created using teacher-student machine learning paradigm," in *Proceedings of the 19th International Society for Music Information Retrieval Conference (ISMIR 2018)*, pp. 431-437, 2018.

[5] C.-L. Hsu and J.-S. R. Jang, "On the Improvement of Singing Voice Separation for Monaural Recordings Using the MIR-1K Dataset," *IEEE Trans. Audio, Speech, and Language Processing*, volume 18, issue 2, pp. 310-319, 2010.

[6] M. Ryynanen and A. Klapuri, "Transcription of the Singing Melody in Polyphonic Music," in *Proceedings of the 7th International Society for Music Information Retrieval Conference (ISMIR 2006)*, pp.222-227, October 2006.

[7] W. Krige, T. Herbst, and T. Niesler, "Explicit transition modelling for automatic singing transcription," *Journal of New Music Research*, vol. 37, no. 4, pp. 311–324, 2008.

[8] L. Yang, A. Maezawa, J. B. L. Smith and E. Chew, "Probabilistic transcription of sung melody using a pitch dynamic model," *2017 IEEE International Conference on Acoustics, Speech and Signal (ICASSP)*, pp. 301-305, 2017.

[9] R. Nishikimi, E. Nakamura, S. Fukayama, M. Goto and K. Yoshii, "Automatic Singing Transcription Based on Encoder-decoder Recurrent Neural Networks with a Weakly-supervised Attention Mechanism," *2019 IEEE International Conference on Acoustics, Speech and Signal (ICASSP)*, 2019.

[10] Fu Z.-S. and L. Su, "Hierarchical Classification Networks for Singing Voice Segmentation and Transcription," in *Proceedings of the 20th International Society for Music Information Retrieval Conference (ISMIR 2019)*, pp. 900-907, 2019.

[11] R. Hennequin, A. Khlif, F. Voituret, and M. Moussallam, "Spleeter: A Fast and State-of-the Art Music Source Separation Tool with Pre-Trained Models," Late-Breaking/Demo ISMIR 2019.

[12] J. Schlüter and S. Böck, "Musical Onset Detection with Convolutional Neural Networks," in *Proceedings of the 6th International Workshop on Machine Learning and Music*, 2013.

[13] J.-C. Chen, J.-S. R. Jang, "TRUES: Tone Recognition Using Extended Segments," ACM Transactions on Asian Language Information Processing, No. 10, Vol. 7, Aug 2008.

[14] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inform. Theory*, vol. IT–13, pp. 260–269, 1967.

[15] M. Mauch, C. Cannam, R. Bittner, G. Fazekas, J. Salamon, J. Dai, J. Bello, and S. Dixon, "Computer-aided melody note transcription using the tony software: Accuracy and efficiency," in *Proceedings of the 1st International Conference on Technologies for Music Notation and Representation*, 2015.

[16] M. Tan, Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," *International Conference on Machine Learning (ICML)*, 2019.

[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, 2015.

[18] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC Music Database: Popular, Classical, and Jazz Music Databases," in *Proceedings of the 3rd International Conference on Music Information Retrieval (ISMIR 2002)*, pp.287-288, October 2002.