ADTWEB: AN OPEN SOURCE BROWSER BASED AUTOMATIC DRUM TRANSCRIPTION SYSTEM

Carl Southall, Nicholas Jillings, Ryan Stables, Jason Hockman
DMT Lab, Birmingham City University
Birmingham, United Kingdom
{carl.southall, nicholas.jillings, ryan.stables, jason.hockman}@bcu.ac.uk

ABSTRACT

This project presents a web based automatic drum transcription (ADT) system called ADTWeb, which improves on existing systems by automatically generating drum tabulature and enabling web-scale user access. ADTWeb consists of two main stages: i) ADT and ii) monitoring and improvement. The transcription uses pre-trained soft attention mechanism neural networks for each of the three drum instruments under observation (i.e., kick drum, snare drum and hi-hat), to determine drum onset locations. The monitoring stage is where user interaction is tracked and users can contribute additional audio and annotations to improve the ADT process.

1. INTRODUCTION

Automatic drum transcription (ADT) systems identify and classify drum onsets in an audio signal. A high-performing open source ADT system would remove the time-consuming manual transcription process in the projects of musicians and researchers. Currently, the only open source ADT systems [1] [2] require programming knowledge to implement, and at best only output onset times as a list in a text file. To enable a larger audience to have access to a state-of-the-art ADT system, we present ADTWeb—a browser-based ADT system—which requires no download or installation and can automatically generate drum tabulatures. To date, recurrent neural network ADT systems have demonstrated to be the state-of-the-art in ADT evaluations [2, 4] although the results may not fully reflect their performance in real world cases due to the limitations of the evaluation data. As ADTWeb is based in a browser, it is possible to track user interactions with the system towards a more holistic understanding of system performance. Finally, ADTWeb also enables users to submit additional audio and annotations to aid ADT improvement.

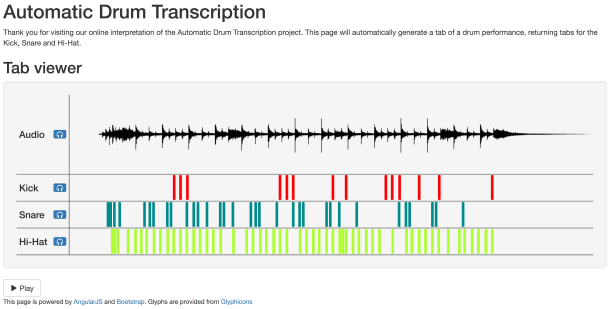

Figure 1. Screenshot of a completect transcription.

2. OVERVIEW

An overview of ADTWeb [3] is given in Figure 2. The tool is made up of two parts: submission and editing. The submission page allows users to submit audio files for analysis and optionally allow the audio to be stored afterwards. This is shown in the top left of the diagram. The editing phase allows users to view the transcription on a free-form timeline, as shown in Figure 1 and outlined in Section 2.2.

2.1 Automatic Drum Transcription

For ADT within the submission stage, the automatic creation of drum tabulature requires two sub-processes: 1) the identification and classification of drum onsets, and 2) the placing of the drum onsets into musical time. This is achieved using client and server side operations within the ADTW framework. First, on the client-side the user drops audio into the browser, then this audio is transferred to the server where it is fed into a pre-trained ADT system similar to the SA system proposed in [3]. A spectrogram is derived from the input audio and processed by three soft attention mechanism neural networks—one for each drum instrument under observation—containing long short-term memory cells with peephole connections. To determine the temporal locations of the drum onsets, peak-picking is performed on the activation functions output by the networks. Next, in order to place drum onsets into musical time the tempo and time signature of the musical piece are required.

[1] https://github.com/CarlSouthall/ADTLib
[2] https://github.com/cwu307/NmfDrumToolbox

[3] https://dmtlab.bcu.ac.uk/ADT/

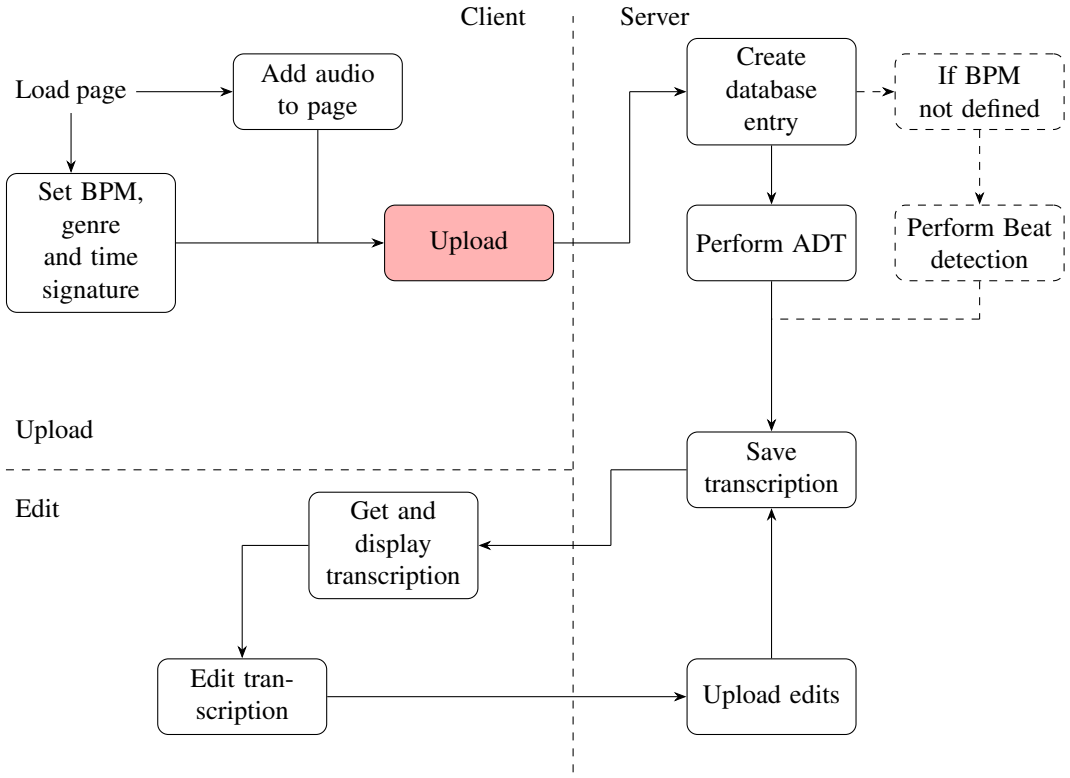

Figure 2. Overview of ADTW with client server interactions, highlighting the two user based states for audio upload and transcription editing.

These can either be provided by the user or may be extracted automatically using the DBNDownBeatTracker algorithm from the madmom python library [1]. Finally, a drum tabulature is generated on the client-side after the required information has been transferred back from the server and is available for download. The ADT stage is the same as the system in the ADTLib python package.[3]

2.2 Monitoring and Improvement

Each transcription is added into a database, along with a timestamp and a fingerprint of the audio file used for the transcription. This is returned to the user and presented as a graphical interface, as in Figure 1. The audio is shown along with a marker for each detected onset for the three drum instruments under observation. Drum sound events can be auditioned with audio playback to confirm the accuracy of the ADT detection. The user can then modify the transcript by adding, moving and deleting detected onsets. These edits are tracked and stored in the database.

The user interaction provides several key benefits. The performance of the ADTLib can be directly evaluated; difficult passages may be highlighted and used in targeted retraining of the system. A web-based ADT system also facilitates the emergence of a community-sourced database, in which users can freely review each others edits and augment entries.

3. REFERENCES

[1] Sebastian Böck, Filip Korzeniowski, Jan Schlüter, Florian Krebs, and Gerhard Widmer. madmom: A new Python audio and music signal processing library. In *Proceedings of the ACM International Conference on Multimedia*, pages 1174–1178, Amsterdam, The Netherlands, 2016.

[2] Carl Southall, Ryan Stables, and Jason Hockman. Automatic drum transcription using bi-directional recurrent neural networks. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 591–597, New York City, United States, 2016.

[3] Carl Southall, Ryan Stables, and Jason Hockman. Automatic drum transcription for polyphonic recordings using soft attention mechanisms and convolutional neural networks. In *Proceedings of the International Society for Music Information Retrieval Conference (ISMIR)*, pages 606–612, Suzhou, China, 2017.

[4] Richard Vogl, Matthias Dorfer, and Peter Knees. Drum transcription from polyphonic music with recurrent neural networks. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 201–205, New Orleans, Louisiana, United States, March 2017.