University of Bergen
Department of Informatics

Automatic Drum Transcription using Deep Learning

Author: Runar Fosse
Supervisor: Pekka Parviainen

Add correct NT faculty, instead of MAT-NAT

UNIVERSITETET I BERGEN
Det matematisk-naturvitenskapelige fakultet

March, 2025

Abstract

Write proper abstract

Lorem ipsum dolor sit amet, his veri singulis necessitatibus ad. Nec insolens periculis ex. Te pro purto eros error, nec alia graeci placerat cu. Hinc volutpat similique no qui, ad labitur mentitum democritum sea. Sale inimicus te eum.

No eros nemore impedit his, per at salutandi eloquentiam, ea semper euismod meliore sea. Mutat scaevola cotidieque cu mel. Eum an convenire tractatos, ei duo nulla molestie, quis hendrerit et vix. In aliquam intellegam philosophia sea. At quo bonorum adipisci. Eros labitur deleniti ius in, sonet congue ius at, pro suas meis habeo no.

Acknowledgements

Est suavitate gubergren referrentur an, ex mea dolor eloquentiam, novum ludus suscipit in nec. Ea mea essent prompta constituam, has ut novum prodesset vulputate. Ad noster electram pri, nec sint accusamus dissentias at. Est ad laoreet fierent invidunt, ut per assueverit conclusionemque. An electram efficiendi mea.

Write proper acknowledgements

`Runar Fosse`
Wednesday 12th March, 2025

Contents

Chapter 1

Introduction

Within the field of Music Information Retrieval (MIR), the task of Automatic Music Transcription (AMT) is considered to be, both an important, and challenging research problem. It describes the process of generating a symbolic notation from audio. The majority of instruments are melodic, where key information for transcription would be to discern pitch, onset time, and duration. This stands in contrast to percussive instruments, where instead of pitch and duration one would focus on instrument classification and onset detection. This sets the stage for Automatic Drum Transcription (ADT), which is a subfield of AMT, specifically focusing on drums and percussive instruments. [7]

Previously, a popular approach to ADT was using signal processing, which later developed into using classical machine learning methods [7]. However in later years, deep learning has shown to be quite effective. In recent years, the focus of most authors has therefore been to find the best performing deep learning approaches by either constructing and analysing the best performing model architectures, or by finding datasets which allow models to generalize the best. [8]

Provide a good introduction into the master thesis, mentioning AMT, ADT and why deep learning is suited for such a task.

1.1 Thesis statement

This leads us to two primary questions. Which deep learning architecture is the best suited for solving a task like this? And, what makes a dataset optimal by making models generalize?

These are the two questions we will try to answer in this thesis. In addition, we will also analyse two standard approaches when it comes to ADT and see how effective they really are. These are, usage of log-filtered spectrograms, and frequency-based, dynamic timestep loss-weighting during training.

Present the aim of the thesis here. And the questions! How do we train a model capable of solving such a task at a high performing level. More specifically:

What architectures are suited for learning such a task? What datasets / combination of datasets makes the model generalize best? Of the many techniques made to help models learn this task, which ones actually help? (Ablation)

Remember the concrete What do we want to figure out.

Chapter 2

Background

2.1 Automatic Drum Transcription

As mentioned, ADT describes the task of transcribing symbolic notation for drums from audio. To be even more descriptive, ADT can be split into further tasks. From least to most complex we have: Drum Sound Classification (DSC), where we classify drum instruments from isolated recordings. Drum Transcription of Drum-only Recordings (DTD), where we transcribe audio containing exclusively drum instruments. Drum Transcription in the Presence of Additional Percussion (DTP), where we transcribe audio containing drum instruments, and additional percussive instruments which the transcription should exclude. Finally, we have Drum Transcription in the Presence of Melodic Instruments (DTM), which describes the task of drum transcription with audio containing both drum, and melodic instruments. [7]

In this thesis, we will focus on the most complex of these, namely DTM. Intuitively, we want to develop a deep learning model which, given input audio, has the ability to detect and classify different drum instrument onsets (events), while selectively ignoring unrelated, melodic instruments.

This task comes with difficulties not seen in the less complex tasks. Zehren et al. [8] describes one example, in where *"melodic and percussive instruments can overlap and mask eachother..., or have similar sounds, thus creating confusion between instruments"*.

Even though deep learning has shown to be the most promising method to solve this task, different approaches has been tried, many with great success. Vogl et al. [6, 5]

showed good results with both a convolutional, and a convolutional-recurrent neural network. Zehren et al. [8, 9] focused on datasets, showing that the amount of data and quality of data are equally important to get good performance. Most recently, Chang et al. [3] explored an autoregressive, language model approach. This approach explored multi-instrument transcriptions, but their results on ADT were notable.

This reinforces the fact that there still exist many approaches to attempt, which could lead to a general improvement on ADT models.

Mention some more detailed background into ADT, DTM, etc. Mention some more details into how it started, how it is going, etc. Also why this is of interest. (Maybe also what is missing)

Mention the sequence to sequence prediction.

2.2 Audio

The US army [1] early described sound as *"the sensation caused in the nervous system by vibration of the delicate membranes of the ear."*. In short, sound is the human perception of acoustic waves in a transition medium, like air. These waves, consisting of vibrating molecules, get picked up by our auditory organs and perceived by the brain.

Thus sound can be described as the propogation and perception of waves. Mathematically, waves can be studied as signals [2]. To represent these sounds digitally, as *audio*, one can express these waves as a signal, giving rise to the *waveform*. The waveform is a representation of a signal as a graph, and charts the amplitude, or strength of the signal, over time.

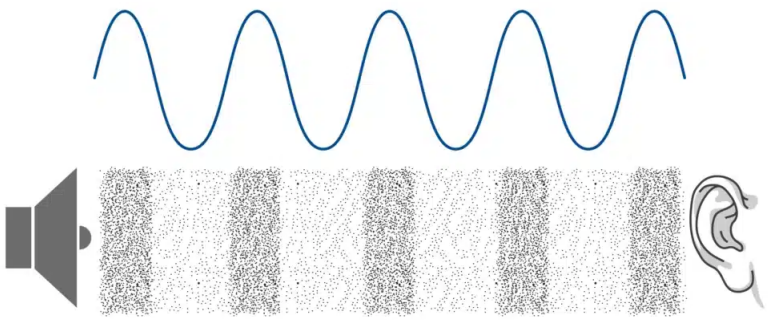

Figure 2.1: Soundwave to waveform relationship

For monophonic sound, this waveform is a one-dimensional representation (in contrast to stereophonic, which is two-dimensional). Even though this is an excellent way of storing audio digitally, it is very compact. There have been deep learning models working directly with these waveforms, e.g. Oord et al.'s WaveNet [4], however the task of just parsing and perceiving such a signal is a complex one.

2.2.1 Fourier Transform

By utilising the fact that

Chapter 3

Datasets

Mention the different already existing datasets used.

3.1 ENST

3.2 MDB

3.3 EMG-D

3.4 Slakh

3.5 ADTOF-YT

Chapter 4

Methods

4.1 Task

Precisely explain the task we are solving. Explain what the input data is, what the labels are. Give intuition into what exactly we want our model to predict.

Here we also explain the input and output, i.e. the data and the labels. What do they look like in their un-preprocessed form?

Here we can also give a figure into the pipeline itself, for better intuition.

4.2 Pipeline

Talk about the general ADT pipeline.

4.2.1 Preprocessing

Now explain what we do to the data before prediction. Explain why we use log-spectrograms / log-mel-spectrograms and how they are computed. Shortely explain what they mean, intuition. Explain the preprocessing step we do afterwards.

And explain how we preprocess the labels (target widening, etc.).

Figures?

4.2.2 Training

Mention the loss function used, and why we use this (BCEWithLogitsLoss).

Mention the computation of infrequency weights, i.e. how they are computed, why they are computed, the intuition into how they will help us...

4.2.3 Postprocessing

Mention how model outputs a "confidence in event happening" distribution, which we want to discretize into events. I.e. explain Vogl's peak picking algorithm [6].

Need some figures here.

4.2.4 Performance Measures

Mention how F-measure (F1-score) is the most used and why. Compare this to accuracy, balanced accuracy. Mention precision, recall and their meaning.

Mention the difference in class-wise, micro- and macro-F1, and why we choose to focus on class-wise and micro in this thesis. Also mention how these are all computed.

Lastly, mention what denotes a True Positive. Mention why we allow a window for a prediction to be correct.

Figures!

4.3 Architectures

Mention the different architectures trained in the first task. Mention something about why we chose them, (e.g. have they shown to be promising in the past? In other fields?).

Figures!

4.4 Datasets

Talk about the different datasets used, how they differ and why we use them.

4.5 Experiments

Here we mention the setups for each of the experiments.

Mention that we use RayTune to train, with PyTorch models. Mention that we only used RayTune's FIFOScheduler, and how for random search / grid search we used their built in parameter space functionality.

Mention that every single experiment was trained for at most 100 epochs, with a early stopping if validation loss didn't decrease within 10 epochs. Mention the learning rate scheduler, where we reduce the learning rate by a factor of 5 if the model hasn't improved in the last 3 epochs.

Mention that we perform early stopping on the validation loss (and why, like the smooth nature, overfitting prevention, etc.), where as we store the best performing model based on the validation F1-score (due to this representing overall prediction performance).

Mention that every experiment is model selected using hold-out validation, and best model is chosen based on micro F1-score.

4.5.1 Architecture experiment

Shortly mention what we do, what the goal is, what we want to figure out.

Architectures

Mention the different architectures trained, and at what hyperparameters they were trained over.

Datasets

Mention the different datasets used, and tested over.

4.5.2 Dataset generalization experiment

Shortly mention why, what, like in the previous experiment.

Architectures

Mention which architectures we now use, and why we chose them. (And hyperparameters)

Datasets

Now mention which datasets / combination of datasets we use. Mention how we now use zero-shot testing (and maybe why).

4.5.3 Ablation experiments?

Technique 1

Technique 2

Technique 3

Chapter 5

Results

5.1 Architecture experiment

Display a table of results, class-wise and micro-F1: Best archicecture per dataset is bolded.

	Dataset 1	Dataset 2	Dataset 3	Dataset 4
CNN	0.5			
RNN	0.4			
Conv-RNN	0.8			
Conv-Attention	0.9			
Transformer	**0.95**			

Display the results in a barplot, to easily capture well-performing models.

Also plot enough information to be able to conclude about overall performance of models, performance on rarer instruments, etc.

5.2 Dataset generalization experiment

Display a table of results, possibly both class-wise and micro-F1 (or maybe just micro-F1): Zero-shot tests have a grayed background, best zero-shot test are bolded.

	Dataset 1	Dataset 2	Dataset 3	Dataset 4
Dataset 1	0.5	0.3		
Dataset 2	0.4	0.8		
Dataset 1+2	0.8	0.7		
Dataset 3	**0.9**	0.6		
Dataset 1+2+3	0.95	0.8		
Dataset 1+4	0.95	**0.75**		
Dataset 1+2+3+4	0.95	0.82		

Display the results in a barplot, to easily capture well-performing models.

Also plot enough information to be able to conclude about overall performance of models, performance on rarer instruments, etc.

5.3 Ablation experiments

Display results and data to be able to conclude if techniques help training / give better end results.

I.e., do we converge faster? Do we converge to a better minimum?

Could plot some loss over epochs? Need to be thorough (or average) to ensure that gains/losses are due to technique (and not hyperparameter choice, etc.).

List of Acronyms and Abbreviations

ADT Automatic Drum Transcription.

AMT Automatic Music Transcription.

DSC Drum Sound Classification.

DTD Drum Transcription of Drum-only Recordings.

DTM Drum Transcription in the Presence of Melodic Instruments.

DTP Drum Transcription in the Presence of Additional Percussion.

MIR Music Information Retrieval.

Bibliography

[1] *Fundamentals of Telephony*. United States, Department of the Army, 1953.
URL: https://books.google.no/books?id=8nvJ6qvtdPUC.

[2] Pragnan Chakravorty. What is a signal? [lecture notes]. *IEEE Signal Processing Magazine*, 35(5):175–177, 2018. doi: 10.1109/MSP.2018.2832195.

[3] Sungkyun Chang, Emmanouil Benetos, Holger Kirchhoff, and Simon Dixon. Yourmt3+: Multi-instrument music transcription with enhanced transformer architectures and cross-dataset stem augmentation, 2024.
URL: https://arxiv.org/abs/2407.04822.

[4] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio, 2016.
URL: https://arxiv.org/abs/1609.03499.

[5] Richard Vogl, Matthias Dorfer, Gerhard Widmer, and Peter Knees. Drum transcription via joint beat and drum modeling using convolutional recurrent neural networks. In *International Society for Music Information Retrieval Conference*, 2017.
URL: https://api.semanticscholar.org/CorpusID:21314796.

[6] Richard Vogl, Gerhard Widmer, and Peter Knees. Towards multi-instrument drum transcription, 2018.
URL: https://arxiv.org/abs/1806.06676.

[7] Chih-Wei Wu, Christian Dittmar, Carl Southall, Richard Vogl, Gerhard Widmer, Jason Hockman, Meinard Müller, and Alexander Lerch. A review of automatic drum transcription. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(9):1457–1483, 2018. doi: 10.1109/TASLP.2018.2830113.

[8] Mickaël Zehren, Marco Alunno, and Paolo Bientinesi. High-quality and reproducible automatic drum transcription from crowdsourced data. *Signals*, 4(4):768–787, 2023.

ISSN 2624-6120. doi: 10.3390/signals4040042.
URL: https://www.mdpi.com/2624-6120/4/4/42.

[9] Mickaël Zehren, Marco Alunno, and Paolo Bientinesi. Analyzing and reducing the synthetic-to-real transfer gap in music information retrieval: the task of automatic drum transcription, 2024.
URL: https://arxiv.org/abs/2407.19823.