Performance MIDI To Score Automatic Transcription

Mateusz Szymański MP.SZYMANSK3@STUDENT.UW.EDU.PL
University of Warsaw, Poland

Abstract

The goal of this project is to explore and understand the latest advancements in the automatic transcription of performance MIDI streams into musical scores, specifically within the broader scope of Audio Music Transcription (AMT). We analyze the recent paper *Performance MIDI-to-score Conversion by Neural Beat Tracking* (Liu et al., 2022), which comprises two main components: rhythm quantization and score generation based on a Convolutional Recurrent Neural Network (CRNN) architecture.

We investigate the model's behavior using sequence perturbations and a LIME-like approach.

1. Introduction

Performance MIDI-to-score transcription is a subset of the broader task of converting raw audio signals into musical scores, known as Automatic Music Transcription (AMT).

This task typically involves several phases:

1. **Conversion of the raw audio signal to a spectrogram:** A spectrogram visually represents audio with two dimensions: time and frequency, and color intensity indicating the amplitude of a particular frequency.

2. **Conversion of the spectrogram to a MIDI stream:** A MIDI stream records musical events, such as the start and end of notes, without being divided into bars.

3. **Conversion of the MIDI stream to a score:** Notes are quantized to specific lengths, the stream is divided into bars, and key signature, time signature, and clef assignments are made.

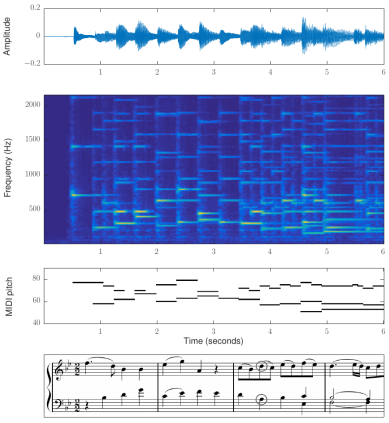

Figure 1: Intermediate AMT steps:
1. raw audio signal,
2. spectrogram,
3. MIDI stream,
4. score.
Source: Benetos et al., 2019.

We focus on examining a model addressing the final phase, specifically designed for piano compositions where each track corresponds to either the left or right hand.

2. Model

The model comprises two main components: rhythm quantization and score generation. Our primary interest lies in the latter. It assigns various musical elements, including tempo,

downbeats, beats, musical onsets, note duration, time signature numerator, time signature denominator, key signature, and hand parts.

The hand part model, key signature model, and time signature models can be considered independent, with no information flow between them. Refer to Figure 2 for a detailed model architecture.

The model takes a MIDI stream as input, encoded as $\mathbf{x} = \{(p_n, o_n, d_n, v_n)\}_{n=1}^{N}$ of shape $(N, 4)$, with N rows encoding N notes. The features include *pitch* p_n (from 0 to 127), *onset* o_n (in seconds), *duration* d_n (in seconds), and *velocity* v_n (an integer from 0 to 127).

We utilized the model trained by the authors of the paper.

3. Ceteris Paribus

We conducted a *ceteris paribus* analysis focusing on musical element models, making several assumptions:

- Note velocity should not affect time signature or hand part assignment.
- Note duration should not impact key signature.
- Note pitch is crucial for key signature/hand part assigning, while other features should not contribute.
- Note pitch does not matter for the time signature.

These should be regarded as general guidelines rather than strict principles.

3.1 Perturbations

For a sample $M = 50$ musical pieces from the dataset, we introduced perturbations to test the assumptions:

- Changing velocity with standard deviations σ_v of 8, 16, 32, and 64.
- Scaling note lengths by a factor from an interval $\alpha \in (\alpha_l, 1)$, where $\alpha_l \in \{0.9, 0.75, 0.5, 0.2\}$ (note extension may lead to overlapping).
- Changing pitch with standard deviations σ_p of 12 and 24 (whole octave, double octave).

Each transformation is applied separately to each feature, note by note. Notice that we don't change onsets as they change the musical structure of a piece.

For each feature model f: *hand part* f_H, *key signature* f_K, *time signature* f_T, we calculated the average error between the output sequence and the output of a perturbed sequence.

For each element $\mathbf{x}_i$ in the dataset sample $i \in \{1, 2, \ldots, 50\}$ and for each change, we sampled $m = 10$ perturbations $\mathbf{x}^{(j)}$ of the original item and measured the error:

$$\text{error}\left(f\right) = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{m}\sum_{j=1}^{m}\text{error}\left(f\left(\mathbf{x}_i\right), f\left(\mathbf{x}_i^{(j)}\right)\right)$$

Here, error between two sequences $\mathbf{x} = (x_i)_{i=1}^{N}$ and $\mathbf{y} = (y_i)_{i=1}^{N}$ is defined as:

$$\text{error}(\mathbf{x}, \mathbf{y}) = \frac{1}{N}\sum_{i=1}^{N}\left[x_i \neq y_i\right]$$

Higher error values indicate greater influence on the output f from a transformation.

This analysis directly measures the model's robustness to specific transformations and does not rely on the model's overall performance.

3.2 Results

Among all three models, the key signature f_K model proved to be robust to all proposed transformations, with an average error of less than 1.5% in each category.

The hand part model f_H error is robust to note shortening (maximum average error of 3%), but inconsistent when note velocities are changed. This inconsistency is mitigated by maintaining consistent velocity for notes played simultaneously. We hypothesize that chords played by one hand typically have similar velocities (see Table 2 for detailed results).

The time signature model is quite robust to note velocity and pitch manipulations (average errors less than 13%), but loses consistency when notes are shortened.

	velocity change σ_v				**duration change α_l**				**pitch change σ_p**	
model	8	16	32	64	0.9	0.75	0.50	0.20	12	24
f_H	7.61	15.65	25.69	34.80	0.29	0.76	1.67	3.08		
f_K	0.07	0.15	0.32	0.46	0.12	0.32	0.75	1.13		
f_T	1.05	2.10	3.78	9.77	3.42	10.10	25.21	39.67	6.50	12.73

Table 1: The average errors of certain perturbations (in percent).

	velocity change σ_v			
variant	8	16	32	64
standard	7.61	15.65	25.69	34.80
uniform within groups	2.70	6.44	11.50	15.14

Table 2: The average errors for the hand part model f_H for 1. standard perturbation, 2. uniform random change for notes played in the same time. The second transformation introduces less inconsistencies.

Figure 3 in the Appendix illustrates all results, including interaction perturbations.

3.3 Local Feature Importance

We encountered challenges with common explainable machine learning tools:

- Input data is a tensor of variable length, not conforming to tabular or image-like formats supported by many tools.
- The input data is a very high-dimensional space, computationally infeasible for certain methods (e.g. Shapley values).
- Pitch perturbations introduce non-integer values, and the pitch space lacks a meaningful measure of distance.

- Some model components (e.g. GRU blocks, ELU activation function) are not supported by certain XAI libraries.

We developed a custom solution to overcome these obstacles. For velocity, we applied a LIME-like approach due to the absence of metric structure in the pitch space. We generated a locally modified version of the MIDI stream tensor for each note, randomizing the velocity for one note. We created 100 samples for each note, calculating model predictions to compare with the original prediction. This approach aids in explaining the findings from the previous section (see Figure 5 for an example).

We also measured the mean influence through this approach: the square root of the mean of squares of linear regression coefficients. This approach aligns with the *ceteris paribus* analysis.

model	f_H	f_K	f_T
average influence	**0.37**	0.04	0.05

Table 3: The average influence of three models calculated by a LIME-like approach.

However, the proposed approach ignores the feature interaction and silently assumes variable independence, both horizontally and vertically. This is a serious limitation of this approach.

3.4 Key Signature Assignment by Note Omission

For the key signature model, as the pitch space is not a metric space, we applied a different strategy. The method is as follows:

1. Get the key signature prediction values (before the last activation function) for the entire sequence.
2. Individually remove each note in a sequence and calculate the difference between the original prediction and the prediction made on a sequence without a single note.
3. Compute a contribution score, represented by the mean of the differences for each note.

This approach enables the measurement of each note's contribution to the attribution of a specific scale. Refer to Figure 6 for an illustrative example.

4. Conclusion

We were able to discover certain artifacts of the score generation models, especially when it comes to velocity contribution to hand part assignment. The time signature model is not fully robust to alterations that should not have affect the output.

Unfortunately, the proposed methods have drawbacks and are not fully justified. Current XAI methods do not work well with symbolic music data in general. Developing more tailored and adaptable XAI methods for musical applications could contribute to improved model interpretability. One of the challenge would be to find a reasonable (and interpretable) embedding of the pitch space that encodes musical features.

References

Emmanouil Benetos, Simon Dixon, Zhiyao Duan, and Sebastian Ewert. Automatic music transcription: An overview. *IEEE Signal Process. Mag.*, 36(1):20–30, 2019. doi: 10.1109/MSP.2018.2869928. URL https://doi.org/10.1109/MSP.2018.2869928.

Francesco Foscarin, Andrew McLeod, Philippe Rigaux, Florent Jacquemard, and Masahiko Sakai. ASAP: a dataset of aligned scores and performances for piano transcription. In *International Society for Music Information Retrieval Conference (ISMIR)*, pages 534–541, 2020.

Bernd Krueger. Classical piano midi, 1996. URL http://www.piano-midi.de.

Lele Liu, Qiuqiang Kong, Veronica Morfi, and Emmanouil Benetos. Performance midi-to-score conversion by neural beat tracking. In Preeti Rao, Hema A. Murthy, Ajay Srinivasamurthy, Rachel M. Bittner, Rafael Caro Repetto, Masataka Goto, Xavier Serra, and Marius Miron, editors, *Proceedings of the 23rd International Society for Music Information Retrieval Conference, ISMIR 2022, Bengaluru, India, December 4-8, 2022*, pages 395–402, 2022. URL https://archives.ismir.net/ismir2022/paper/000047.pdf.

Adrien Ycart and Emmanouil Benetos. A-maps: Augmented maps dataset with rhythm and key annotations. In *19th International Society for Music Information Retrieval Conference, ISMIR, Late Breaking and Demos Papers.*, 2018. URL https://qmro.qmul.ac.uk/xmlui/handle/123456789/45985.

Appendix A. Model

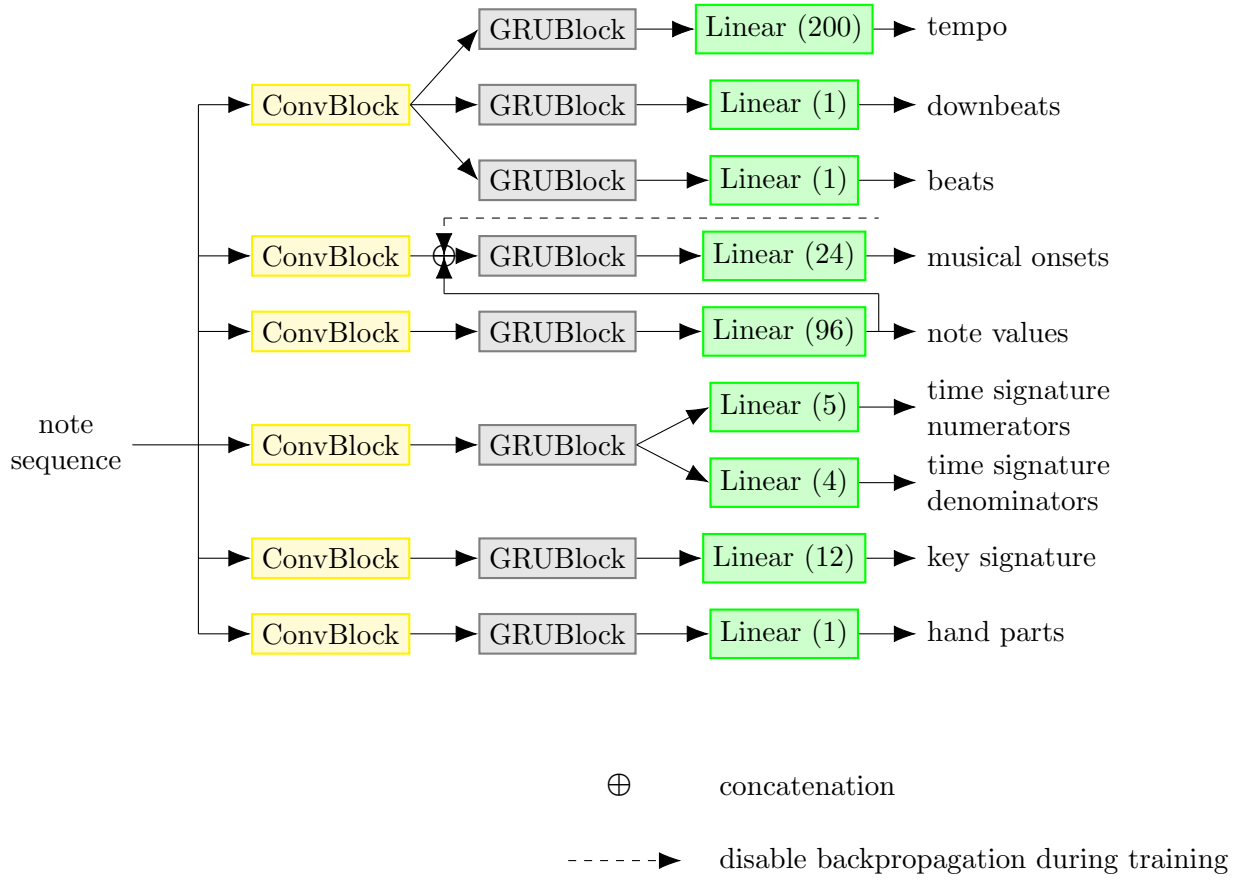

Figure 2: An architecture of the model. There are six separate modules of the entire model in total: `beat`, `quantization`, `time signature`, `key signature` and `hand part`.

Appendix B. Dataset

The model has been trained on three datasets containing classical piano compositions, including Bach, Mozart, Beethoven, Chopin or Liszt.

As the authors of the original work, for model analysis we used three different datasets:

- A-MAPS v1.1 (Ycart and Benetos (2018)), 266 items
- ASAP (Foscarin et al. (2020)), 1586 items
- CPM (Krueger (1996)), 337 items

We took a sample of $M = 50$ pieces to work with.

Appendix C. Results

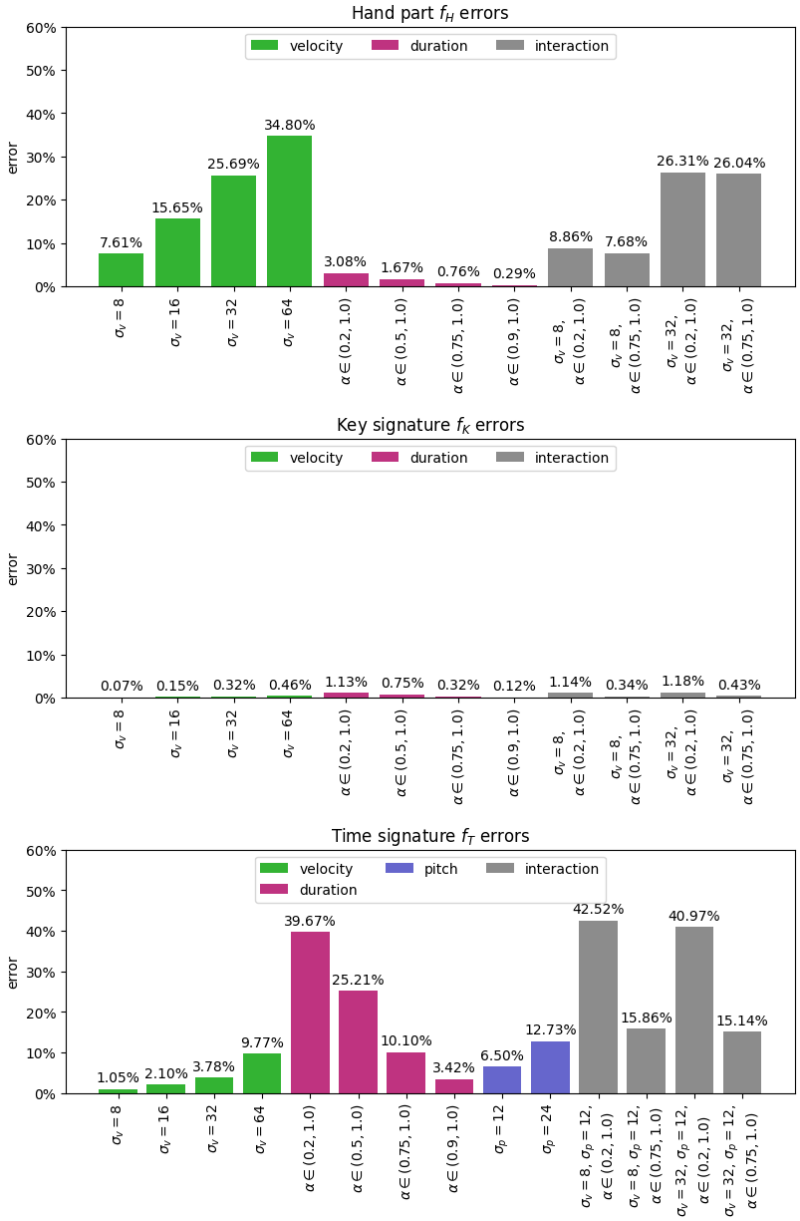

Figure 3: Results of the *ceteris paribus* experiment for the hand part model f_H, the key signature model f_K and the time signature model f_T. The hand part model is robust to note duration changes while it is susceptible to velocity manipulation. On the other hand, the time signature model is sensitive to note duration changes, which is expected to some extent, and relatively robust to other transformations. The key signature model is robust to all perturbations.

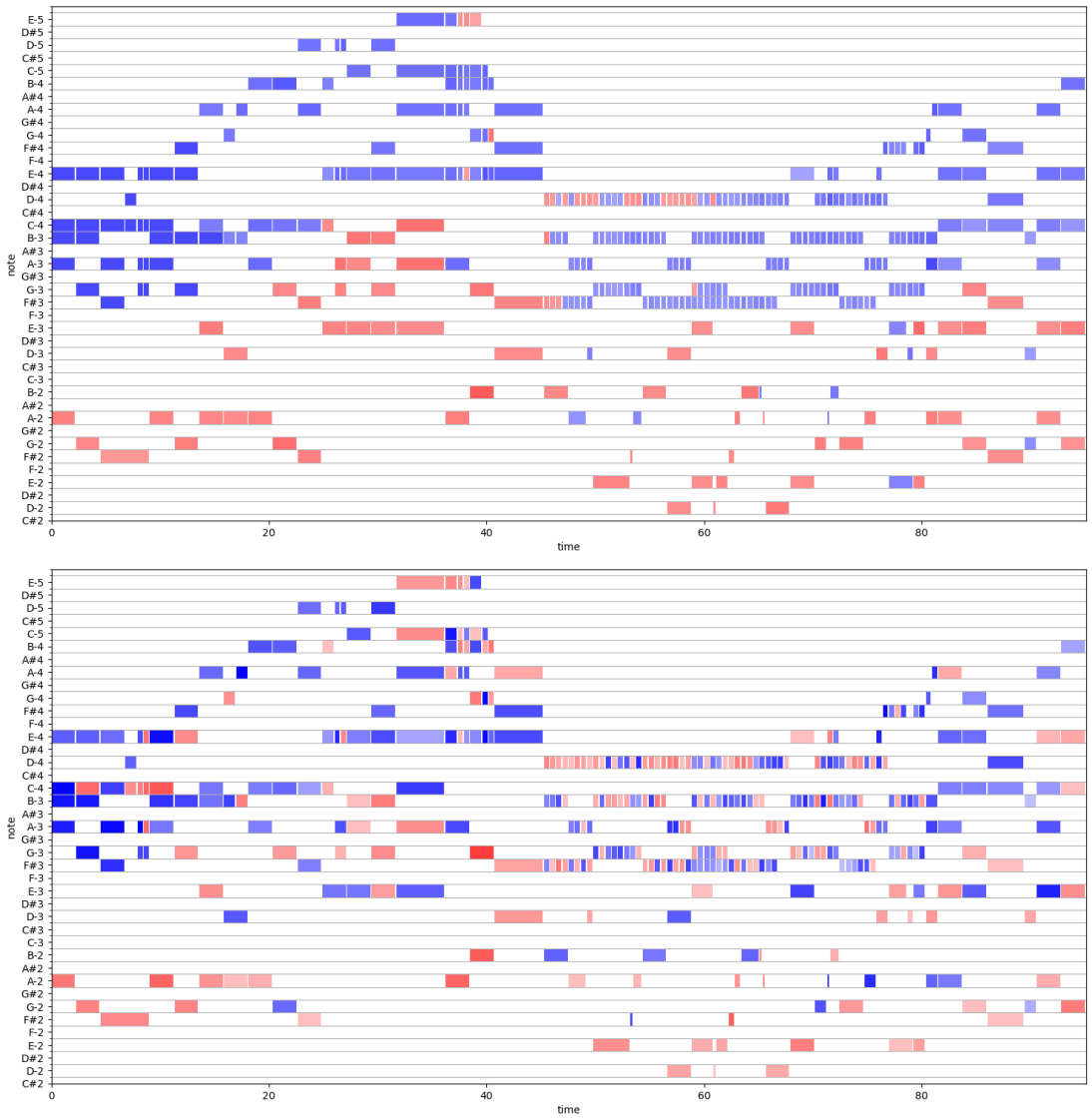

Figure 4: The piano roll for the hand part model output. Each note is represented as a (blue or red) block, indicating a pitch (note), duration and velocity. The first graph represents an original sequence while the second has perturbed velocity. Left-hand notes are marked as red. We can see that there are much many misalignment in the hand assignment in the second output. As a rule of thumb, the left hand notes (red) should be at the bottom while the right hand ones (blue) should be at the top.

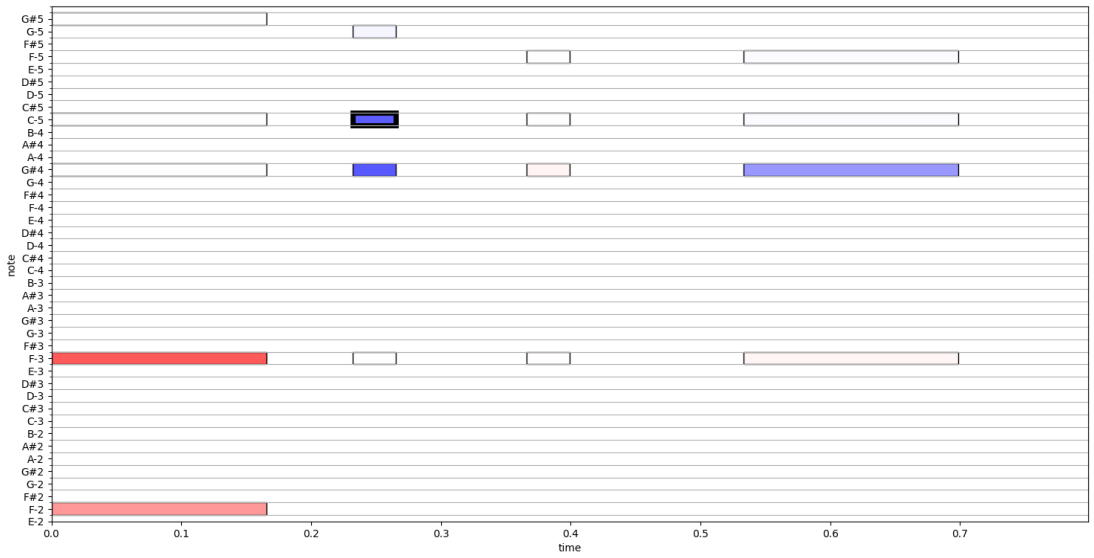

Figure 5: A graph showing how the velocity of the second $C5$ note (with a thick outline) influences hand part assignments of other notes. It can be observed that the current velocity of this note makes the model think of the first $F3$ note as a left hand note. Reducing the note $C5$ velocity to a low value results in a misassigning $F3$ as the right hand note. This is, of course, a undesired behavior of the model.

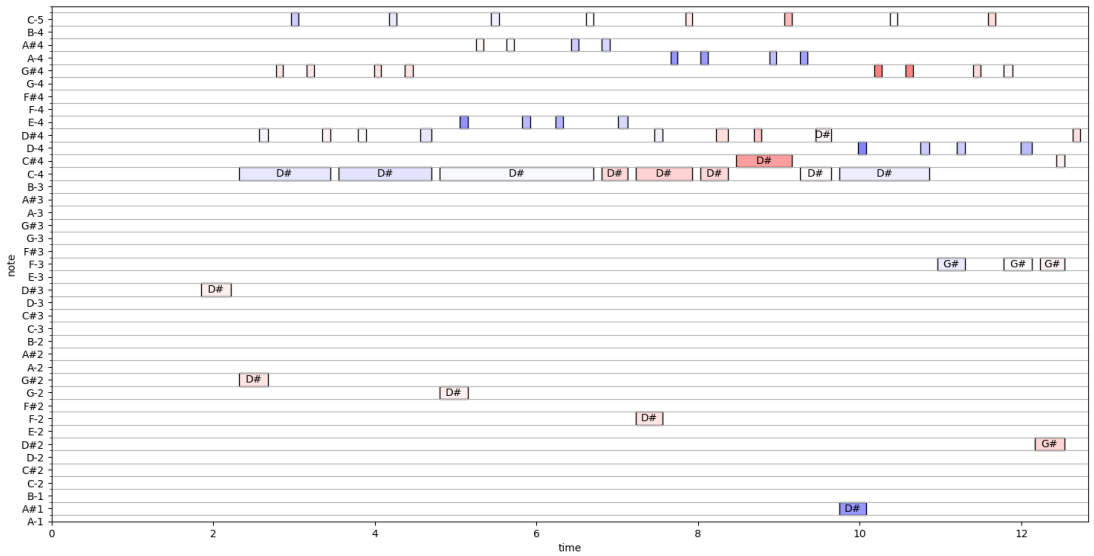

Figure 6: A key signature alignment for $D\sharp$ scale. The names on the notes represent the model key signature attribution. There are two different scales assigned by the model: $D\sharp$ scale and $G\sharp$ scale. Notes: D, E and A are not in the scale and have a negative influence on the assignment.