



The GigaMIDI Dataset with Features for Expressive Music Performance Detection

DATASET ARTICLES

KEON JU MAVERICK LEE **JEFF ENS****SARA ADKINS****PEDRO SARMENTO****MATHIEU BARTHET****PHILIPPE PASQUIER**

*Author affiliations can be found in the back matter of this article

]ubiquity press

ABSTRACT

The Musical Instrument Digital Interface (MIDI), introduced in 1983, revolutionized music production by allowing computers and instruments to communicate efficiently. MIDI files encode musical instructions compactly, facilitating convenient music sharing. They benefit music information retrieval (MIR), aiding in research on music understanding, computational musicology, and generative music. The GigaMIDI dataset contains over 1.4 million unique MIDI files, encompassing 1.8 billion MIDI note events and over 5.3 million MIDI tracks. GigaMIDI is currently the largest collection of symbolic music in MIDI format available for research purposes under fair dealing. Distinguishing between non-expressive and expressive MIDI tracks is challenging, as MIDI files do not inherently make this distinction. To address this issue, we introduce a set of innovative heuristics for detecting expressive music performance. These include the distinctive note velocity ratio (DNVR) heuristic, which analyzes MIDI note velocity; the distinctive note onset deviation ratio (DNODR) heuristic, which examines deviations in note onset times; and the note onset median metric level (NOMML) heuristic, which evaluates onset positions relative to metric levels. Our evaluation demonstrates these heuristics effectively differentiate between non-expressive and expressive MIDI tracks. Furthermore, after evaluation, we create the most substantial expressive MIDI dataset, employing our heuristic NOMML. This curated iteration of GigaMIDI encompasses expressively performed instrument tracks detected by NOMML, containing all General MIDI instruments, constituting 31% of the GigaMIDI dataset, totaling 1,655,649 tracks.

CORRESPONDING AUTHOR:

Keon Ju Maverick Lee

Metacreation Lab for Creative AI, School of Interactive Arts and Technology, Simon Fraser University, Vancouver, Canada

keon_maverick@sfu.ca

KEYWORDS:

Symbolic Music, MIDI Dataset, Computational Musicology, Digital Musicology, Musical Features, Expressive Music Performance Analysis

TO CITE THIS ARTICLE:

Lee, K.J.M., Ens, J., Adkins, S., Sarmento, P., Barthet, M., & Pasquier, P. (2025). The GigaMIDI Dataset with Features for Expressive Music Performance Detection. *Transactions of the International Society for Music Information Retrieval*, 8(1), 1–19.DOI: <https://doi.org/10.5334/tismir.203>

1 INTRODUCTION

The representation of digital music can be categorized into two main forms: audio and symbolic domains. Audio representations of musical signals characterize sounds produced by acoustic or digital sources (e.g., acoustic musical instruments, vocals, found sounds, virtual instruments, etc.) in an uncompressed or compressed way. In contrast, symbolic representation of music relies on a notation system to characterize the musical structures created by a composer or resulting from a performance (e.g., scores, tablatures, MIDI performance). While audio representations intrinsically encode signal aspects correlated to timbre, this is not the case for symbolic representations; however, symbolic representations may refer to timbral identity (e.g., *cello* staff) and expressive features correlated with timbre (e.g., *pianissimo* or *forte* dynamics) through notations.

Multiple encoding formats are employed for the representation of music. WAV is frequently utilized to store uncompressed audio, thereby retaining nuanced timbral attributes. In contrast, MIDI serves as a prevalent format for the symbolic storage of music data. MIDI embraces a multitrack architecture to represent musical information, enabling the generation of a score representation through score editor software. This process encompasses diverse onset timings and velocity levels, facilitating quantification and encoding of these musical events (MIDI Association, 1996a).

The choice of training dataset significantly influences deep learning models, particularly highlighted in the development of symbolic music generation models (Adkins et al., 2023; Briot, 2021; Briot and Pachet, 2020; Brunner et al., 2018; Ens and Pasquier, 2020; Hernandez-Oliván and Beltrán, 2022; Huang et al., 2019; Payne, 2019; von Rütte et al., 2023; Shih et al., 2022). Consequently, MIDI datasets have gained increased attention as one of the main resources for training these deep learning models. Within automatic music generation via deep learning, end-to-end models use digital audio waveform representations of musical signals as input (Dieleman et al., 2018; Manzelli et al., 2018; Zukowski and Carr, 2017). Automatic music generation based on symbolic representations (Raffel and Ellis, 2016b; Zhang, 2020) uses digital notations to represent musical events from a composition or performance; these can be contained, e.g., in a digital score, a tablature (Sarmiento et al., 2023a,b), or a piano-roll. Moreover, symbolic music data can be leveraged in computational musicology to analyze the vast corpus of music using MIR and music data mining techniques (Li et al., 2012).

In computational creativity and musicology, a critical aspect is distinguishing between non-expressive performances, which are mechanical renditions of a score, and expressive performances, which reflect variations that convey the performer's personality and style. MIDI

files are commonly produced through score editors or by recording human performances using MIDI instruments, which allow for adjustments in parameters, such as velocity or pressure, to create expressively performed tracks.

However, MIDI files typically do not contain metadata distinguishing between non-expressive and expressive performances, and most MIR research has focused on file-level rather than track-level analysis. File-level analysis examines global attributes such as duration, tempo, and metadata, aiding structural studies, while track-level analysis explores instrumentation and arrangement details. The note-level analysis provides the most granular insights, focusing on pitch, velocity, and microtiming to reveal expressive characteristics. Together, these hierarchical levels form a comprehensive framework for studying MIDI data and understanding expressive elements of musical performances.

Our work categorizes MIDI tracks into two types: non-expressive tracks, defined by fixed velocities and quantized rhythms (though expressive performances may also exhibit some degree of quantization), and expressive tracks, which feature microtiming variations compared with the nominal duration indicated on the score as well as dynamics variations, translating into velocity changes across and within notes. To address this, we introduce novel heuristics in Section 4 for detecting expressive music performances by analyzing microtimings and velocity levels to differentiate between expressive and non-expressive MIDI tracks.

The main contributions of this work can be summarized as follows: (1) The GigaMIDI dataset, which encompasses over 1.4 million MIDI files and over five million instrument tracks, is introduced. This data collection is the largest open-source MIDI dataset for research purposes to date. (2) We have developed novel heuristics (Heuristic 1 and 2) tailored explicitly for detecting expressive music performance in MIDI tracks. Our novel heuristics were applied to each instrument track in the GigaMIDI dataset, and the resulting values were used to evaluate the expressiveness of tracks in GigaMIDI. (3) We provide details of the evaluation results (Section 5.2) of each heuristic to facilitate expressive music performance research. (4) Through the application of our optimally performing heuristic, as determined through our evaluation, we create the largest MIDI dataset of expressive performances, specifically incorporating instrument tracks beyond those associated with piano and drums (which constitute 31% of the GigaMIDI dataset), totaling over 1.6 million expressively performed MIDI tracks.

2 BACKGROUND

Before exploring the GigaMIDI dataset, we examine symbolic music datasets in existing literature. This sets the stage for our discussion on MIDI's musical expression and

performance aspects, laying the groundwork for understanding our heuristics in detecting expressive music performance from MIDI data.

2.1 SYMBOLIC MUSIC DATA

Symbolic formats refer to the representation of music through symbolic data, such as MIDI files, rather than audio recordings (Zeng et al., 2021). Symbolic music understanding involves analyzing and interpreting music on the basis of its symbolic data, namely information about musical notation, music theory, and formalized music concepts (Simonetta et al., 2018).

Symbolic formats have practical applications in music information processing and analysis. Symbolic music processing involves manipulating and analyzing symbolic music data, which can be more efficient and easier to interpret than lower-level representations of music, such as audio files (Cancino-Chacón et al., 2022).

The Musical Instrument Digital Interface (MIDI) is a technical standard that enables electronic musical instruments and computers to communicate by transmitting event messages that encode information such as pitch, velocity, and timing. This protocol has become integral to music production, allowing for the efficient representation and manipulation of musical data (Meroño-Peñuela et al., 2017). MIDI datasets, which consist of collections of MIDI files, serve as valuable resources for musicological research, enabling large-scale analyses of musical trends and styles. For instance, studies utilizing MIDI datasets have explored the evolution of popular music (Mauch et al., 2015) and facilitated advancements in music transcription technologies through machine learning techniques (Qiu et al., 2021). The application of MIDI in various domains underscores its significance in both the creative and analytical aspects of contemporary music.

Symbolic music processing has gained attention in the MIR community, and several music datasets are available in symbolic formats (Cancino-Chacón et al., 2022). Symbolic representations of music can be used for style classification, emotion classification, and music piece matching (Zeng et al., 2021). Symbolic formats also play a role in the automatic formatting of music sheets.

XML-compliant formats, such as the WEDEL format, include constructs describing integrated music objects, including symbolic music scores (Bellini et al., 2005). Besides that, the Music Encoding Initiative (MEI) is an open, flexible format for encoding music scores in a machine-readable way. It allows for detailed representation of musical notation and metadata, making it ideal for digital archiving, critical editions, and musicological research (Crawford and Lewis, 2016).

ABC notation is a text format used to represent music symbolically, particularly favored in folk music (Cros Vila and Sturm, 2023). It offers a human-readable method for notating music, with elements represented using letters, numbers, and symbols. This format is easily learned, written, and converted into standard notation or MIDI files using software, enabling convenient sharing and playback of musical compositions (Figure 1).

Csound notation, part of Csound software, symbolically represents electroacoustic music (Licata, 2002). It controls sonic parameters precisely, fostering complex compositions blending traditional and electronic elements. This enables innovative experimentation in contemporary music. Max Mathews' MUSIC 4, developed in 1962, laid the groundwork for Csound, introducing key musical concepts to computing programs.

With the proliferation of deep learning approaches, often driven by the need for vast amounts of data, the creation and curation of symbolic datasets have been active in this research area. The MIDI format can be considered the most common music format for symbolic music datasets, despite alternatives such as the Essen folk music database in ABC format (Schaffrath, 1995), JSB Chorales Dataset available via MusicXML format and Music21 (Boulanger-Lewandowski et al., 2012; Cuthbert and Ariza, 2010), and Guitar Pro tablature format (Sarmiento et al., 2021).

Focusing on MIDI, Table 1 showcases symbolic music datasets. MetaMIDI (Ens and Pasquier, 2021) is a collection of 436,631 MIDI files. MetaMIDI comprises a substantial collection of multi-track MIDI files primarily derived from an extensive music corpus characterized by longer duration. Approximately 57.9% of MetaMIDI includes a piece that has a drum track.

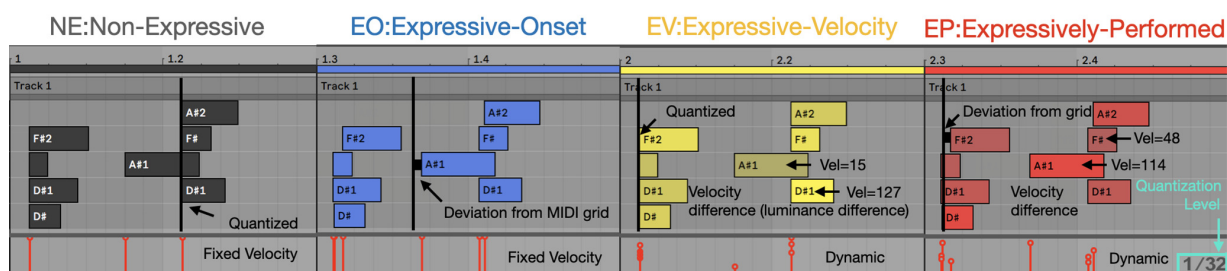


Figure 1 Four classes (NE = non-expressive, EO = expressive onset, EV = expressive velocity, and EP = expressively performed) using heuristics in Section 4.2 for the expressive performance detection of MIDI tracks in GigaMIDI.

Dataset	Format	Files	Hours	Instruments
GigaMIDI	MIDI	>1.43M	>40,000	Misc.
MetaMIDI	MIDI	436,631	>20,000	Misc.
Lakh MIDI	MIDI	174,533	>9,000	Misc.
DadaGP	Guitar Pro	22,677	>1,200	Misc.
ATEPP	MIDI	11,677	1,000	Piano
Essen Folksong	ABC	9,034	56.62	Piano
NES Music	MIDI	5,278	46.1	Misc.
MID-FILD	MIDI	4,422	>40	Misc.
MAESTRO	MIDI	1,282	201.21	Piano
Groove MIDI	MIDI	1,150	13.6	Drums
JSB Chorales	MusicXML	382	>4	Misc.

Table 1 Sample of symbolic datasets in multiple formats, including MIDI, ABC, MusicXML, and Guitar Pro formats.

ATEPP = Automatically Transcribed Expressive Piano Performances.

The Lakh MIDI dataset (LMD) encompasses a collection of 174,533 MIDI files (Raffel, 2016), and an audio-to-MIDI alignment matching technique (Raffel and Ellis, 2016a) is introduced, which is also utilized in MetaMIDI for matching musical styles if scraped style metadata is unavailable.

2.2 MUSIC EXPRESSION AND PERFORMANCE REPRESENTATIONS OF MIDI

We use the terms “expressive MIDI,” “human-performed MIDI,” and “expressive machine-generated MIDI” interchangeably to describe MIDI files that capture expressively performed (EP) tracks, as illustrated in Figure 1. EP-class MIDI tracks capture performances by human musicians or producers, emulate the nuances of live performance, or are generated by machines trained with deep learning algorithms. These tracks incorporate variations of features, such as timing, dynamics, and articulation, to convey musical expression.

From the perspective of music psychology, analyzing expressive music performance involves understanding how variations of, e.g., timing, dynamics, and timbre (Barthet et al., 2010) relate to performers’ intentions and influence listeners’ perceptions. Repp’s research demonstrates that expressive timing deviations, such as rubato, enhance listeners’ perception of naturalness and musical quality by aligning with their cognitive expectations of flow and structure (Repp, 1997b). Palmer’s work

further reveals that expressive timing and dynamics are not random but rather result from skilled motor planning, as musicians use mental representations of music to execute nuanced timing and dynamic changes that reflect their interpretive intentions (Palmer, 1997).

Our focus lies on two main types of MIDI tracks: non-expressive and expressive. Non-expressive MIDI tracks exhibit relatively fixed velocity levels and onset deviations, resulting in metronomic and mechanical rhythms. In contrast, expressive MIDI tracks feature subtle temporal deviations (non-quantized but humanized or human-performed) and greater variations in velocity levels associated with dynamics.

2.2.1 Non-expressive and expressively performed MIDI tracks

MIDI files are typically produced in two ways (excluding synthetic data from generative music systems): using a score/piano roll editor or recording a human performance. MIDI controllers and instruments, such as a keyboard and pads, can be utilized to adjust the parameters of each note played, such as velocity and pressure, to produce expressively performed MIDI. Being able to distinguish non-expressive and expressive MIDI tracks is useful in MIR applications. However, MIDI files do not accommodate such distinctions within their metadata. MIDI-track-level analysis for music expression has received less attention from MIR researchers than MIDI-file-level analysis. Previous research regarding interpreting MIDI velocity levels (Dannenberg, 2006) and modeling dynamics/expression (Berndt and Hähnel, 2010; Ortega et al., 2019) was conducted, and a comprehensive review of computational models of expressive music performance is available in (Cancino-Chacón et al., 2018). Generation of expressive musical performances using a case-based reasoning system (Arcos et al., 1998) has been studied in the context of tenor saxophone interpretation and the modeling of virtuosic bass guitar performances (Goddard et al., 2018). Velocity prediction/estimation using deep learning was introduced at the MIDI note level (Collins and Barthet, 2023; Kim et al., 2022; Kuo et al., 2021; Tang et al., 2023).

2.2.2 Music expression and performance datasets

The aligned scores and performances (ASAP) dataset has been developed specifically for annotating non-expressive and expressively performed MIDI tracks (Foscarin et al., 2020). Comprising 222 digital musical scores synchronized with 1,068 performances, ASAP encompasses over 92 hours of Western classical piano music. This dataset provides paired MusicXML and quantized MIDI files for scores, along with paired MIDI files and partial audio recordings for performances. The alignment of ASAP includes annotations for downbeat, beat, time signature, and key signature, making it notable for its incorporation of music scores aligned with MIDI and

audio performance data. The MID-FiLD (Ryu et al., 2024) dataset is the sole dataset offering detailed dynamics for Western orchestral instruments. However, it primarily focuses on creating expressive dynamics via MIDI Control Change #1 (modulation wheel) and lacks velocity variations, featuring predominantly constant velocities, as verified by our manual inspection. In contrast, the GigaMIDI dataset focuses on expressive performance detection through variations of micro-timings and velocity levels.

The MAESTRO (Hawthorne et al., 2019) and Groove MIDI (Gillick et al., 2019) datasets focus on singular instruments, specifically piano and drums, respectively. Despite their narrower scope, these datasets are noteworthy for including MIDI files exclusively performed by human musicians. Saarland Music Data (SMD) contains piano performance MIDI files and audio recordings, but SMD only contains 50 files (Müller et al., 2011). The Vienna 4 × 22 Piano Corpus (Goebel, 1999) and the Batik-plays-Mozart Corpus MIDI dataset (Hu and Widmer, 2023) both provide valuable resources for studying classical piano performance. The Vienna 4 × 22 Piano Corpus features high-resolution recordings of 22 pianists performing four classical pieces with the aim of analyzing expressive elements such as timing and dynamics across performances. Meanwhile, the Batik-plays-Mozart dataset offers MIDI recordings of Mozart pieces performed by the pianist Batik, capturing detailed performance data such as note timing and velocity. Together, these datasets support research in performance analysis and machine learning applications in music.

The Automatically Transcribed Expressive Piano Performances (ATEPP) dataset (Zhang et al., 2022) was devised for capturing performer-induced expressiveness by transcribing audio piano performances into MIDI format. ATEPP addresses inaccuracies inherent in the automatic music transcription process. Similarly, the GiantMIDI-Piano dataset (Kong et al., 2022), akin to ATEPP, comprises artificial intelligence (AI)-transcribed piano tracks that encapsulate expressive performance nuances. However, we excluded the ATEPP and GiantMIDI-Piano datasets from our expressive music performance detection task. State-of-the-art transcription models are known to overfit the MAESTRO dataset (Edwards et al., 2024) due to its recordings originating from a controlled piano competition setting. These performances, all played on similar Yamaha Disklavier pianos under concert hall conditions, result in consistent acoustic and timbral characteristics. This uniformity restricts the models' ability to generalize to out-of-distribution data, contributing to the observed overfitting.

3 GIGAMIDI DATA COLLECTION

We present the GigaMIDI dataset in this section and its descriptive statistics, such as the MIDI instrument group, the number of MIDI notes, ticks per quarter note, and

musical style. Additional descriptive statistics are in Supplementary file 1: Appendix (A.1).

3.1 OVERVIEW OF THE GIGAMIDI DATASET

The GigaMIDI dataset is a superset of the MetaMIDI dataset (Ens and Pasquier, 2021), and it contains 1,437,304 unique MIDI files with 5,334,388 MIDI instrument tracks, and 1,824,536,824 (over 10^9 ; hence, the prefix “Giga”) MIDI note events. The GigaMIDI dataset includes 56.8% single-track and 43.2% multi-track MIDI files. It contains 996,164 drum tracks and 4,338,224 non-drum tracks. The initial version of the dataset consisted of 1,773,996 MIDI files. Approximately 20% of the dataset was subjected to a cleaning process, which included deduplication achieved by verifying and comparing the MD5 checksums of the files. While we integrated certain publicly accessible MIDI datasets from previous research endeavors, it is noteworthy that over 50% of the GigaMIDI dataset was acquired through web-scraping and organized by the authors.

The GigaMIDI dataset includes per-track loop detection, adapting the loop detection and extraction algorithm presented in (Adkins et al., 2023) to MIDI files. In total, 7,108,181 loops with lengths ranging from 1 to 8 bars were extracted from GigaMIDI tracks, covering all types of MIDI instruments. Details and analysis of the extracted loops from the GigaMIDI dataset will be shared in a companion paper report via our GitHub page.

3.2 COLLECTION AND PREPROCESSING OF THE GIGAMIDI DATASET

The authors manually collected and aggregated the GigaMIDI dataset, applying our heuristics for MIDI-based expressive music performance detection. This aggregation process was designed to make large-scale symbolic music data more accessible to music researchers.

Regarding data collection, we manually gathered freely available MIDI files from online sources such as Zenodo,¹ GitHub,² and public MIDI repositories by web scraping. The source links for each subset are provided via our GitHub webpage.³ During aggregation, files were organized and deduplicated by comparing MD5 hash values. We also standardized each subset to the General MIDI (GM) specification, ensuring coherence; for example, non-GM drum tracks were remapped to GM. Manual curation was employed to assess the suitability of the files for expressive music performance detection, with particular attention to defining ground truth tracks for expressive and non-expressive categories. This process involved systematically identifying the characteristics of expressive and non-expressive MIDI track subsets by manually checking the characteristics of MIDI tracks in each subset. The curated subsets were subsequently analyzed and incorporated into the GigaMIDI

dataset to facilitate the detection of expressive music performance.

To improve accessibility, the GigaMIDI dataset has been made available on the Hugging Face Hub. Early feedback from researchers in music computing and MIR indicates that this platform offers better usability and convenience compared with alternatives such as GitHub and Zenodo. This platform enhances data preprocessing efficiency and supports seamless integration with workflows, such as MIDI parsing and tokenization using Python libraries such as Symusic⁴ and MidiTok⁵ (Fradet et al., 2021), as well as deep learning model training using Hugging Face. Additionally, the raw metadata of the GigaMIDI dataset is hosted on the Hugging Face Hub⁶ (see Section 8).

As part of preprocessing GigaMIDI, single-track drum files allocated to MIDI channel 1 are subjected to re-encoding. This serves the dual purpose of ensuring their accurate representation on MIDI channel 10, drum channel, while mitigating the risk of misidentification as a piano track, denoted as channel 1. Details of MIDI channels are explained in Section 3.3.1.

Furthermore, all drum tracks in the GigaMIDI dataset were standardized through remapping on the basis of the General MIDI (GM) drum mapping guidelines (MIDI Association, 1996b) to ensure consistency. Detailed information about the drum remapping process can be accessed via GitHub. In addition, the distribution of drum instruments, categorized and visualized by their relative frequencies, is presented in Appendix A.1 (Gómez-Marín et al., 2020).

3.3 DESCRIPTIVE STATISTICS OF THE GIGAMIDI DATASET

3.3.1 MIDI instrument group

The GigaMIDI dataset is divided into three primary subsets: “all-instrument-with-drums,” “drums-only,” and “no-drums.” The “all-instrument-with-drums” subset comprises 22.78% of the dataset and includes multi-track MIDI files with drum tracks. The “drums-only” subset makes up 56.85% of the dataset, containing only drum tracks, while the “no-drums” subset (20.37%) consists of both multi-track and single-track MIDI files without drum tracks. As shown in Figure 2, drums-only files typically have a high-density distribution and are mostly under 50 bars, reflecting their classification as drum loops. Conversely, multi-track and single-track piano files exhibit a broader range of durations, spanning 10–300 bars, with greater diversity in musical structure.

MIDI instrument groups, organized by program numbers, categorize instrument sounds. Each group corresponds to a specific program number range, representing unique instrument sounds. For instance, program numbers 1 to 8 on MIDI Channel 1 are associated with the piano instrument group (acoustic piano, electric piano, harpsichord, etc). The analysis in Table 2

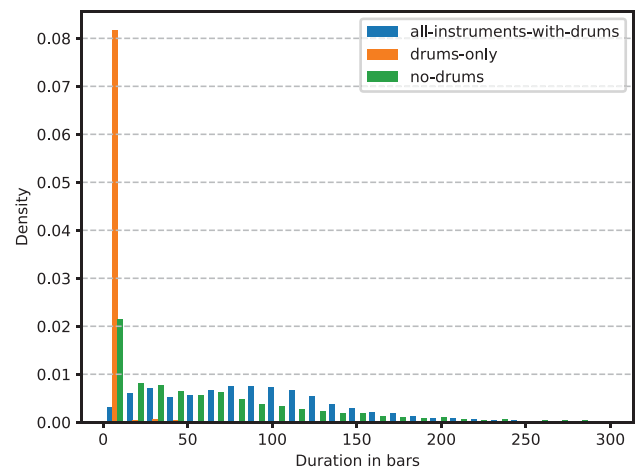


Figure 2 Distribution of the duration in bars of the files from each subset of the GigaMIDI dataset. The x-axis is clipped to 300 for better readability.

IGN: 1-8	Events	IGN: 9-16	Events
Piano	60.2%	Reed/Pipe	1.1%
CP	2.4%	Drums	17.4%
Organ	1.8%	Synth Lead	0.5%
Guitar	6.7%	Synth Pad	0.6%
Bass	4.2%	Synth FX	0.3%
String	1.1%	Ethnic	0.3%
Ensemble	2.1%	Percussive FX	0.3%
Brass	0.7%	Sound FX	0.3%

Table 2 Number of MIDI note events by instrument group in percentage (IGN = instrument group number, CP = chromatic percussion, and FX = effect).

focuses on the occurrence of MIDI note events across the 16 MIDI instrument groups (MIDI Association, 1996b). Channel 10 is typically reserved for the drum instrument group.

Although MIDI groups/channels often align with specific instrument types in the General MIDI specification (MIDI Association, 1996a), composers and producers can customize instrument number allocations based on their preferences.

The GigaMIDI dataset analysis reveals that most MIDI note events (77.6%) are found in two instrument groups: piano and drums. The piano instrument group has more MIDI note events (60.2%) because most piano-based tracks are longer. The higher number of MIDI notes in piano tracks compared to other instrumental tracks can be attributed to several factors. The inherent nature of piano playing, which involves ten fingers and frequently includes simultaneous chords due to its dual-staff layout,

naturally increases note density. Additionally, the piano's wide pitch range, polyphonic capabilities, and versatility in musical roles allow it to handle melodies, harmonies, and accompaniments simultaneously. Piano tracks are often used as placeholders or sketches during composition, and MIDI input is typically performed using a keyboard defaulting to a piano timbre. These characteristics, combined with the cultural prominence of the piano and the practice of condensing multiple parts into a single piano track for convenience, result in a higher density of notes in MIDI datasets.

The GigaMIDI dataset includes a significant proportion of drum tracks (17.4%), which are generally shorter and contain fewer note events compared to piano tracks. This is primarily because many drum tracks are designed for drum loops and grooves rather than for full-length musical compositions. The supplementary file provides a detailed distribution of note events for drum sub-tracks, including each drum MIDI instrument in the GigaMIDI dataset. Sound effects, including breath noise, bird tweets, telephone rings, applause, and gunshot sounds, exhibit minimal usage, accounting for only 0.249% of the dataset. Chromatic percussion (2.4%) stands for pitched percussions, such as glockenspiel, vibraphone, marimba, and xylophone.

3.3.2 Number of MIDI notes and ticks per quarter note

Figure 3(a) shows the distribution for the number of MIDI notes in GigaMIDI. According to our data analysis, the span from the 5th to the 95th percentile covers 13 to 931 notes, indicating a significant presence of short-length drum tracks or loops.

Figure 3(b) illustrates the distribution of ticks per quarter note (TPQN). TPQN is a unit that measures the resolution or granularity of timing information. Ticks are the smallest indivisible units of time within a MIDI sequence. A higher TPQN value means more precise timing information can be stored in a MIDI sequence. The most common TPQN values are 480 and 960. According to our data analysis of GigaMIDI, common TPQN values range from 96 to 960 between the 5th and 95th percentiles.

3.3.3 Musical style

We provide the GigaMIDI dataset with metadata regarding musical styles. This includes our manually curated style metadata by listening to and annotating MIDI files on the basis of the Musicmap style topology (Crauwels, 2016), displayed in Figure 4. We organized all the musical style metadata from our subsets, including remapping drumming styles (Gillick et al., 2019) and DadaGP (Sarmiento et al., 2021) to Musicmap style topology. The acquisition of scraped style metadata, encompassing audio-text match style metadata sourced from the MetaMIDI subset (Ens and Pasquier, 2021), is conducted. Subsequently, all gathered musical style metadata undergoes conversion, adhering to the Musicmap topology for consistency.

The distribution of musical style metadata in the GigaMIDI dataset, illustrated in Figure 5, is based on the Musicmap topology and encompasses 195,737 files annotated with musical style metadata. Notably, prevalent styles include classical, pop, rock, and folk music. These 195,737 style annotations mostly originate from a combination of scraped metadata acquired online, style data present in our subsets, and manual inspection conducted by the authors.

A major challenge in utilizing scraped style metadata from the MetaMIDI subset is ensuring its accuracy of metadata. To address this, a subset of the GigaMIDI dataset, consisting of 29,713 MIDI files, was carefully reviewed through music listening and manually annotated with style metadata by a doctoral-level music researcher.

MetaMIDI integrates scraped style metadata and associated labels obtained through an audio-MIDI matching process.⁷ However, our empirical assessment, based on manual auditory analysis of musical styles, identified inconsistencies and unreliability in the scraped metadata from the MetaMIDI subset (Ens and Pasquier, 2021). To address this, we manually remapped 9,980 audio-text-matched musical style metadata entries within the MetaMIDI subset, ensuring consistent and accurate musical style classifications. Finally, these remapped musical styles were aligned with

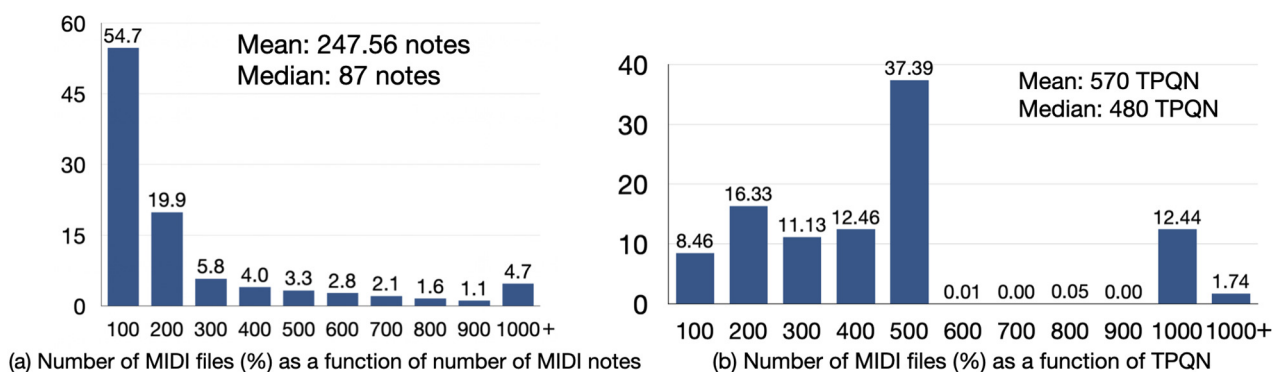


Figure 3 Distribution of files in GigaMIDI according to (a) MIDI notes, and (b) ticks per quarter note (TPQN).

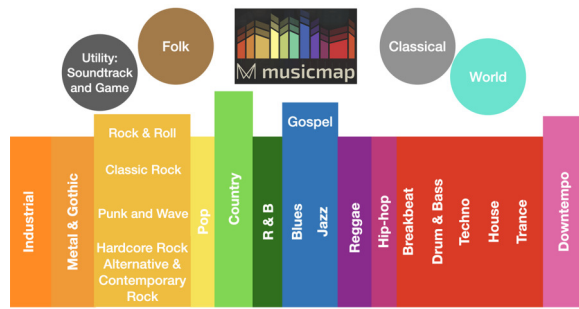


Figure 4 Musicmap style topology (Crauwels, 2016).

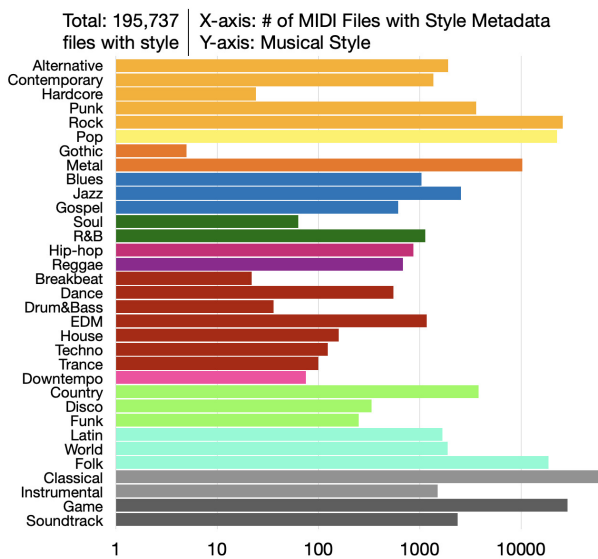


Figure 5 Distribution of musical style in GigaMIDI.

the Musicmap topology to provide more uniform and reliable information on musical style.

We provide the audio-text-matched musical style metadata available using three musical style metadata: Discogs,⁸ Last.fm,⁹ and Tagtraum,¹⁰ collected using the MusicBrainz¹¹ database.

4 HEURISTICS FOR MIDI-BASED EXPRESSIVE MUSIC PERFORMANCE DETECTION

Our heuristic design centers on analyzing variations in velocity levels and onset-time deviations from a metric grid. MIDI velocity replicates the hammer velocity in acoustic pianos, where the force applied to the keys determines the speed of the hammers, subsequently affecting the energy transferred to the strings and, consequently, the amplitude of the resulting vibrations. This concept is integrated into MIDI keyboards, which replicate hammer velocity by using MIDI velocity levels to control the dynamics of the sound. A velocity value of 0 produces no sound, while 127 indicates maximum intensity. Higher velocity values yield louder

notes, while lower ones result in softer tones, analogous to dynamics markings such as *pianissimo* or *fortissimo* in traditional performance. Onset-time deviations in MIDI represent the difference between the actual note timings and their expected positions on a quantized metric grid, with the grid’s resolution being determined by the ticks per quarter note (TPQN) of the MIDI file. These deviations, often introduced through human performance, play a crucial role in conveying musical expressiveness.

The primary objective of our proposed heuristics for expressive performance detection is to differentiate between expressive and non-expressive MIDI tracks by analyzing velocity and onset-time deviations. This analysis is applied at the MIDI track level, with each instrument track undergoing expressive performance detection. Our heuristics, introduced in the following sections, assess expressiveness by examining velocity variations and microtimings, offering a versatile framework suitable for various GM instruments.

Other related approaches for this task are more specific to acoustic piano performance rather than being tailored to MIDI tracks. Key overlap time (Repp, 1997a) and melody lead (Goebel, 2001) focus on acoustic piano performances, analyzing legato articulation and melodic timing anticipation, which limits their application to piano contexts. Similarly, linear basis models (Grachten and Widmer, 2012) focus on Western classical instruments, particularly the acoustic piano, and rely on score-based dynamics (e.g., *crescendo*, *fortissimo*), making them less applicable to non-classical or non-Western music. Such dynamics can be interpreted in MIDI velocity levels, and our heuristics consider this aspect. Compared with these methods, our heuristics offer broader applicability, addressing dynamic variations and microtiming deviations across a wide range of MIDI instruments, making them suitable for detecting expressiveness in diverse musical contexts.

4.1 BASELINE HEURISTIC: DISTINCT NUMBER OF VELOCITY LEVELS AND ONSET-TIME DEVIATIONS

This baseline heuristic focuses solely on analyzing the count of distinct velocity levels (“distinct velocity”) and unique onset-time deviations (“distinct onset”) without considering the MIDI track length. Generally, longer MIDI tracks show more distinct velocities and onset deviations than shorter ones. Designed as a simpler alternative to the more sophisticated Heuristics 1 and 2, this baseline has limited accuracy for MIDI tracks of varying lengths, as it does not adjust for track duration. However, this was not a significant issue during heuristic evaluation in Section 5.2, as most tracks in the evaluation set are longer and have a limited variance in terms of length.

Our baseline heuristic design counts the number of unique velocity levels and onset-time deviations present in a MIDI track. For example, consider a MIDI track where $\mathbf{v} = [64, 72, 72, 80, 64, 88]$ represents the MIDI velocity values, and $\mathbf{o} = [-5, 0, 5, -5, 10, 0]$ represents the onset-time deviations in MIDI ticks. Applying our heuristic, we first store only the unique values in each list: For \mathbf{v} , the distinct velocity levels are $\{64, 72, 80, 88\}$, and for \mathbf{o} , the distinct onset-time deviations are $\{-5, 0, 5, 10\}$. By counting these unique values, we identify four distinct velocity levels and four distinct onset-time deviations for this MIDI track, with no deviation being treated as a specific occurrence.

4.2 DISTINCTIVE NOTE VELOCITY/ONSET DEVIATION RATIO

Distinctive note velocity ratio (DNVR) and distinctive note onset deviation ratio (DNODR) measure the proportion (in a percentage) of unique MIDI note velocities and onset-time deviations in each MIDI track. These metrics form a set of heuristics for detecting expressive performances, classified into four categories: non-expressive (NE), expressive onset (EO), expressive velocity (EV), and expressively performed (EP), as shown in Figure 1. The DNVR metric counts unique velocity levels to differentiate between tracks with consistent velocity and those with expressive velocity variation, while the DNODR calculation helps identify MIDI tracks that are either perfectly quantized or have minimal microtiming deviations.

Heuristic 1 Calculation of Distinctive Note Velocity/Onset Deviation Ratios

- 1: $\mathbf{x} \leftarrow [x_1, \dots, x_n]$ \triangleright list of velocity or onset deviation
- 2: $c_{velocity} \leftarrow 0$ \triangleright number of distinctive velocity levels
- 3: $c_{onset} \leftarrow 0$ \triangleright number of distinctive onset deviations
- 4: **for** $i \leftarrow 2$ **to** n **do** $\triangleright n =$ number of notes in a track
- 5: **if** $x_i \notin \mathbf{x}$ **then**
- 6: $c \leftarrow c + 1$ \triangleright add 1 to c if there is a new value
- 7: **return** $c_{velocity}$ **or** c_{onset}
- 8: $\mathbf{c}_{velocity-ratio} = c_{velocity} \div 127 \times 100$
- 9: $\mathbf{c}_{onset-ratio} = c_{onset} \div TPQN \times 100$

Heuristic 1 is proposed to analyze the variation in velocity levels and onset-time deviations within a MIDI track. Here, $\mathbf{x}_{velocity}$ holds each track's velocity values, while \mathbf{x}_{onset} contains onset deviations from a quantized MIDI grid based on the track's TPQN. For example, a possible set of values could be $x_{velocity} = \{88, 102, \dots\}$ and $x_{onset} = \{-3, 2, 5, \dots\}$, the latter being represented in ticks. The functions $c_{velocity}$ and c_{onset} return the count of unique velocity levels and onset-time deviations per track, respectively. Next, $\mathbf{c}_{onset-ratio}$ is divided by the track's TPQN to represent the proportion of microtiming positions

within each quarter note. Similarly, $\mathbf{c}_{velocity-ratio}$ is divided by 127 (the range of possible velocity levels). Finally, each ratio is converted to a percentage by multiplying by 100.

4.3 MIDI NOTE ONSET MEDIAN METRIC LEVEL

Figure 6 displays the classification of various note onsets into duple metric levels 0–5. Let us define k as the parameter that controls the metric level's depth. The duple onset metric level (**dup**) grid divides the beat into even subdivisions, such as halves or quarters, capturing rhythms in duple meter. The triplet onset metric level (**trip**) grid divides the beat into three equal parts, aligning with triplet rhythms commonly found in swing and compound meters. Notably, since the grey-colored note onset (ML $< \frac{1}{128}$ note metric level) does not belong to any **dup** ^{i} for $0 \leq i \leq 5$, it is assigned to the extra category shown in the bottom row because it is finer than the maximum metric level where $k = 6$. For example, Figure 6 displays the metric level depth. The duple metric level **dup** ^{k} divides each quarter note into 2^k equal pulses, while the triplet metric level **trip** ^{k} divides it into $\frac{3}{2} \times 2^k$ pulses. For our experiments, we choose $k = 6$. Consequently, the maximum metric levels we consider are **dup**⁵ and **trip**⁵, corresponding to the 128th notes. Based on our observation of data in MIDI tracks, this provides a sufficient level of granularity, given the note durations frequently found in most forms of music.

In Heuristic 2, we propose the MIDI note onset median metric level (NOMML), another heuristic for detecting non-expressive and expressively performed MIDI tracks. This heuristic counts the median metric level of note onsets. The metric level $\mathbf{ml}(x)$ for a note onset x is the lowest duple or triplet level that aligns with the onset. Since some pulses overlap between duple and triplet levels, we prioritize duple levels before considering triplets. For instance, with 120 ticks per quarter note, a note onset a at tick 60 aligns with pulses on all metric levels **dup** ^{i} for $i \geq 1$ and **trip** ^{j} for $j \geq 2$. Here, the lowest matching levels are **dup**¹ and **trip**², so by prioritizing duple levels, $\mathbf{ml}(a) = \mathbf{dup}^1$. Conversely, a note onset b at tick 40 aligns only with triplet levels, resulting in $\mathbf{ml}(b) = \mathbf{trip}^1$.

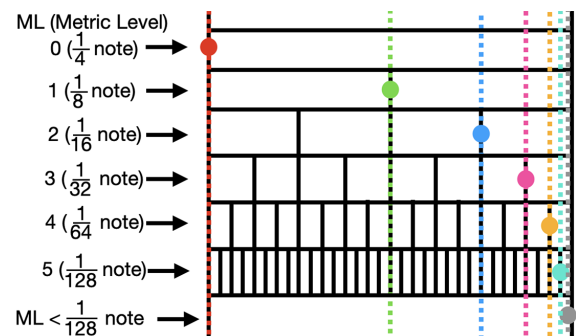


Figure 6 Example of each duple onset metric level grid in different colors using circles and dotted lines for the position of onsets, where $k = 6$.

Heuristic 2 Calculation of Note Onset Median Metric Level

```

1: c ← [] ▷ List of metric levels
2: o ← [ $o_1, \dots, o_n$ ] ▷ List of note onsets (in ticks)
3: TPQN ▷ Ticks per quarter notes of MIDI File
4: for  $i \leftarrow 1$  to  $n$  do ▷ line(4–9): Handle duple onsets
5:   for  $j \leftarrow 0$  to  $k - 1$  do
6:      $p \leftarrow \frac{TPQN}{2^j}$  ▷ periodicity of duple grid
7:     if  $o_i \pmod{p} \equiv 0$  then
8:       c.append( $2j$ ) ▷ multiples of periodicity
9:       break
10:    if  $\|c\| < i$  then ▷ line(10–15): Handle triplet
11:      for  $j \leftarrow 0$  to  $k - 1$  do
12:         $p \leftarrow \frac{2 * TPQN}{3 * 2^j}$  ▷ periodicity of triplet
13:        if  $o_i \pmod{p} \equiv 0$  then
14:          c.append( $2j + 1$ ) ▷ multiples of p
15:          break
16:    if  $\|c\| < i$  then ▷ Handle onsets beyond grid
17:      c.append( $2k$ ) ▷  $k$  = metric level's depth
18: return median(c)

```

Given a list of note onset times (**o**), Heuristic 2 calculates the median metric level. The list **c** is used to store the metric levels for each note onset, so after executing lines 4–17, we have $\mathbf{c} = [\mathbf{ml}(o_1), \dots, \mathbf{ml}(o_n)]$. For example, we have a list of metric levels for note onsets: $c = [2, 3, 4, 6, 3, 7, 8, 3, 4]$. To calculate the median, we first sort c as follows: $c = [2, 3, 3, 3, 4, 4, 6, 7, 8]$. Since the list contains nine values, the median is the middle element, which is the fifth value in the sorted list. Thus, the median metric level for c is 4.

In lines 4–9, the lowest duple metric level is determined for each note onset o_i . The condition in line 10 is met only when o_i does not belong to any duple metric level. Here, $\|c\|$ denotes the current length of **c**. If o_i does not match a duple level, lines 11–15 determine the lowest triplet metric level. When o_i does not belong to any duple or triplet level, it is assigned to an extra category containing both **dup** ^{i} and **trip** ^{i} for any $i \geq k$ (lines 16–17).

To calculate the median metric level, each level is assigned a unique numerical value. Duple and triplet metric levels are interleaved to ensure a meaningful median: duple levels are represented by even numbers (**dup** ^{i} = $2i$) and triplet levels by odd numbers (**trip** ^{i} = $2i + 1$).

5 THRESHOLD AND EVALUATION OF HEURISTICS FOR EXPRESSIVE MUSIC PERFORMANCE DETECTION

Optimal threshold selection involves a structured approach to determine the best threshold for distinguishing between non-expressive (NE) and expressively

performed (EP) tracks. A machine learning regressor aids in identifying this threshold, evaluated using metrics such as classification accuracy and the P4 metric (Sitarz, 2022).

$$P_4 = \frac{4 \cdot TP \cdot TN}{4 \cdot TP \cdot TN + (TP + TN) \cdot (FP + FN)} \quad (1)$$

The selection of the P4 metric (Equation 1; true positives [TP], true negatives [TN], false positives [FP], and false negatives [FN]) over the F1 metric is motivated by the small sample size of ground truths available for non-expressive and expressive tracks in our binary classification task.

The curated set for threshold selection and evaluation is split into 80% training for the threshold selection (Section 5.1) and 20% testing for the evaluation (Section 5.2) to prevent data leakage. Heuristics for expressive music performance detection, described in Section 4, are assessed for classification accuracy on this testing set.

5.1 THRESHOLD SELECTION OF HEURISTICS FOR EXPRESSIVE MUSIC PERFORMANCE DETECTION

The threshold denotes the optimal value delineating the boundary between NE and EP tracks. A significant challenge in identifying the threshold stems from the limited availability of dependable ground-truth instances for NE and EP tracks.

The curation process involves manually inspecting tracks for velocity and microtiming variations to achieve a 100% confidence level in ground truths. Subsets failing to meet this level are strictly excluded from consideration. We selected 361 NE and 361 EP tracks and assigned binary labels 0 for NE and 1 for EP tracks. Our curated set consists of:

1. Non-expressive (361 instances): ASAP (Foscarin et al., 2020) score tracks.
2. Expressively performed (361 instances): ASAP performance tracks, Vienna 4 × 22 Piano Corpus (Goebel, 1999), Saarland Music Data (Müller et al., 2011), Groove MIDI (Gillick et al., 2019), and Batik-plays-Mozart Corpus (Hu and Widmer, 2023).

For the curated set, we intentionally balanced the number of instances across classes to avoid bias. In imbalanced datasets, classification accuracy can be misleadingly high—especially in a two-class setup—where a classifier could achieve high accuracy by predominantly predicting the majority class if one class has significantly more instances (e.g., 10 times more). This bias reduces the model's ability to generalize and perform well on unseen data, especially if both classes are important. As a result, the classification accuracy, precision, and recall metrics can become unreliable, making it difficult to assess the true effectiveness of the heuristics, particularly in detecting or distinguishing the minority class.

To tackle this, balancing the dataset enables a more reliable option for evaluating the classification task, even for baseline heuristics. We partially excluded Groove MIDI and ASAP subsets from the curated set, as if we had included them entirely, the curated set initially would contain roughly 10 times more expressively performed instances than non-expressive ones. A total of 361 instances were selected, as this was the maximum number of non-expressive instances with available ground truth data.

We employ logistic regression (LR; Kleinbaum et al., 2002) alongside leave-one-out cross-validation (LOOCV; Wong, 2015) to determine thresholds using ground truths of NE and EP classes. LR estimates each class probability for binary classification between NE and EP class tracks. LOOCV assesses model performance iteratively by training on all but one data point and testing on the excluded point, ensuring comprehensive evaluation. This is particularly beneficial for small datasets to avoid reliance on specific train-test splits. During this task, the machine learning regressor is solely used for threshold identification rather than classification. The high accuracy of the machine learning regressor facilitates optimal threshold identification without arbitrary threshold selection.

After completing the machine learning classifier's training phase, efforts are directed toward identifying the classifier's optimal boundary point to maximize the P4 metric. However, relying solely on the P4 metric for threshold selection proves inadequate, as it may not comprehensively capture all pertinent aspects of the underlying scenarios.

We manually examine the training set to establish percentile boundaries for distinguishing NE and EP classes based on ground truth data. Specifically, we identify the maximum P4 metric within the 80% training set. Using this boundary range, we determine the optimal threshold index in a feature array that maximizes the P4 metric, which is then used to extract the corresponding threshold for our heuristic. This feature array contains all feature values for each heuristic. The optimal threshold index, selected on the basis of our machine learning regression model and P4 score, identifies the optimal threshold from the feature array. For example, the optimal threshold for the NOMML heuristic is found at level 12, corresponding to the 63.85th percentile, yielding a P4 score of 0.9952, with similar information available for other heuristics in Table 3. Detailed steps for selecting optimal thresholds for each heuristic are provided in Supplementary File: Appendix B.

It is important to note that the analysis in this section is speculative, relying on observations from Tables 4 and 5 without direct supporting evidence at this stage. Later in the evaluation (Section 5.2), we provide corresponding results that substantiate these preliminary insights.

Heuristic	Threshold	P4
Distinct velocity	52	0.7727
Distinct onset	42	0.7225
DNVR	40.965%	0.7727
DNODR	4.175%	0.9529
NOMML	Level 12	0.9952

Table 3 Optimal threshold selection results based on the 80% training set, showing the optimal threshold value for each heuristic where the P4 value is maximized.

Class	Distinct – Onset & Distinct – Velocity
NE (62.5%)	$D - O < 42$ & $D - V < 52$
EO (7.2%)	$D - O \geq 42$ & $D - V < 52$
EV (27.4%)	$D - O < 42$ & $D - V \geq 52$
EP (2.9%)	$D - O \geq 42$ & $D - V \geq 52$

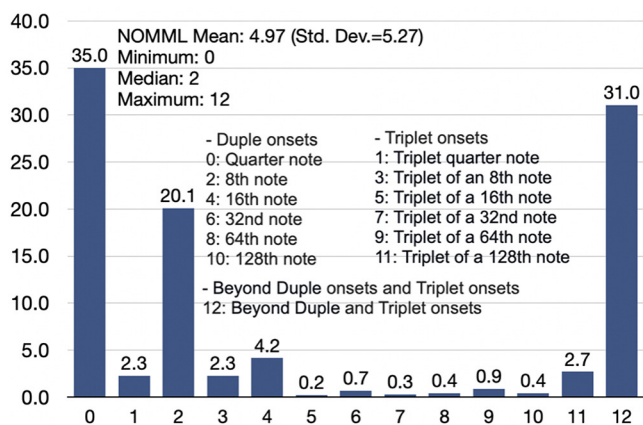
Table 4 Detection results (%) for expressive performance in each MIDI track class within the GigaMIDI dataset.

The analysis is based on the number of distinct velocity levels (D-V = distinct velocity) and onset-time deviations (D-O = distinct onset). Categories include non-expressive (NE), expressive onset (EO), expressive velocity (EV), and expressively performed (EP).

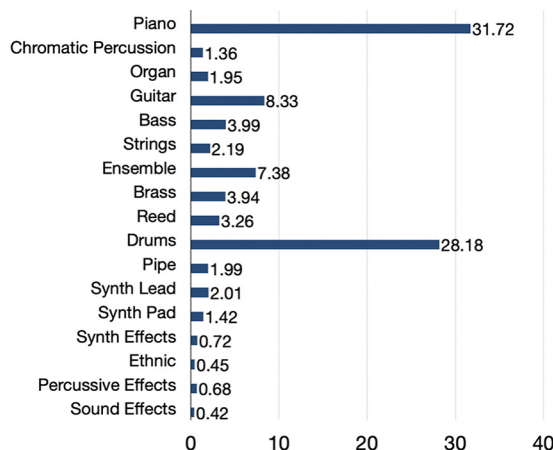
Class	$C_{\text{onset-ratio}(O-R)}$ & $C_{\text{velocity-ratio}(V-R)}$
NE (52.3%)	$C_{O-R} < 4.175\%$ & $C_{V-R} < 40.965\%$
EO (9.1%)	$C_{O-R} \geq 4.175\%$ & $C_{V-R} < 40.965\%$
EV (24.2%)	$C_{O-R} < 4.175\%$ & $C_{V-R} \geq 40.965\%$
EP (14.4%)	$C_{O-R} \geq 4.175\%$ & $C_{V-R} \geq 40.965\%$

Table 5 Results (%) of expressive performance detection for each MIDI track class in GigaMIDI based on the calculation of $C_{\text{onset-ratio}}$ (DNODR), and $C_{\text{velocity-ratio}}$ (DNVR).

Tables 4 and 5 display the distribution of the GigaMIDI dataset across four distinct classes (Figure 1), using optimal thresholds derived from our baseline heuristics (distinct velocity levels and onset-time deviations) and DNVR/DNODR heuristics. With the baseline heuristics (Table 4), class distribution accuracy is limited owing to the prevalence of short-length drum and melody loop tracks in GigaMIDI, which baseline heuristics do not account for. In contrast, results using DNVR/DNODR heuristics (Table 5) show improved class identification, especially for EP and NE tracks, as these heuristics consider MIDI track length, accommodating short loops with



(a) Number of MIDI tracks (%) for each metric level



(b) Number of expressive MIDI tracks (%) for each MIDI instrument group

Figure 7 Distribution of MIDI tracks according to (a) NOMML (level between 0 and 12, where $k = 6$) for MIDI tracks in GigaMIDI. The NOMML heuristic investigates duple and triplet onsets, including onsets that cannot be categorized as duple- or triplet-based MIDI grids, and (b) instruments for expressively performed tracks in the GigaMIDI dataset.

around 100 notes. Although DNVR/DNODR heuristics provide more accurate distributions, both are less robust than the distribution of the NOMML heuristic, as shown in Figure 7(a).

Figure 7(a) illustrates the distribution of NOMML for MIDI tracks in the GigaMIDI dataset. The analysis reveals that the majority of MIDI tracks fall within three distinct bins (bins: 0, 2, and 12), encompassing a cumulative percentage of 86.1%. This discernible pattern resembles a bimodal distribution, distinguishing between NE and EP class tracks.

Figure 7(a) shows 69% of MIDI tracks in GigaMIDI are NE class and 31% of GigaMIDI are EP class tracks (NOMML: 12). Our curated version of GigaMIDI utilizing NOMML level 12 as a threshold is provided. This curated version consists of 869,513 files (81.59% single-track and 18.41% multi-track files) or 1,655,649 tracks (28.18% drum and 71.82% non-drum tracks). The distribution of MIDI instruments in the curated version is displayed in Figure 7(b), indicating that piano and drum tracks are the predominant components.

5.2 EVALUATION OF HEURISTICS FOR EXPRESSIVE PERFORMANCE DETECTION

In our evaluation results (Table 6), the NOMML heuristic clearly outperforms other heuristics, achieving the highest accuracy at 100%. Additionally, onset-based heuristics generally show better accuracy than velocity-based ones. This suggests that distinguishing velocity levels poses a greater challenge. For instance, in the ASAP subset, non-expressive score tracks—encoding traditional dynamics through velocity—display fluctuations rather than a fixed velocity level, whereas these tracks are aligned to a quantized grid, making onset-based detection more straightforward. However, we recognize that

Detection heuristics	Classification accuracy	Ranking
Distinct velocity	77.9%	4
Distinct onset	77.9%	4
DNVR	83.4%	3
DNODR	98.2%	2
NOMML	100%	1

Table 6 Classification accuracy of each heuristic for expressive performance detection.

accuracy alone does not provide a complete understanding, prompting further investigation.

To further investigate, we also report TP, TN, FP, FN, and CN as metrics (shown in Table 7) for assessing the reliability of our heuristics using the optimal thresholds in expressive performance detection, where “true” denotes expressive instances and “false” signifies non-expressive instances. Thus, investigating the capacity to achieve a higher correct-negative ($CN = \frac{TN}{TN + FN}$) rate holds significance in this context, as it assesses the reliable discriminatory power against NE instances, as well as EP instances. As a result, NOMML achieves a 100% CN rate, and other heuristics perform reasonably well.

6 LIMITATIONS

In navigating the use of MIDI datasets for research and creative explorations, it is imperative to consider the ethical implications inherent in dataset bias (Born, 2020). Bias in MIDI datasets often mirrors prevailing practices in Western digital music production, where certain instruments, particularly the piano and drums, as illustrated

Heuristic (%)	TP	TN	FP	FN	CN
Distinct velocity	35.4	42.5	21.2	0.9	98.0
Distinct onset	24.8	53.1	10.6	11.5	82.2
DNVR	35.4	48.0	21.2	0.9	98.2
DNODR	34.5	63.7	0	1.77	97.3
NOMML	36.3	63.7	0	0	100

Table 7 True positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) based on the threshold set by P4 for heuristics, including correct negatives (CN) (in percentages).

in Figure 7(b), dominate. This predominance is largely influenced by the widespread availability and use of MIDI-compatible instruments and controllers for these instruments. The piano is a primary compositional tool and a ubiquitous MIDI controller and keyboard, facilitating input for a wide range of virtual instruments and synthesizers. Similarly, drums, whether through drum machines or MIDI drum pads, enjoy widespread use for rhythm programming and beat production. This prevalence arises from their intuitive interface and versatility within digital audio workstations. This may explain why the distribution of MIDI instruments in MIDI datasets is often skewed toward piano and drums, with limited representation of other instruments, particularly those requiring more nuanced interpretation or those less commonly played via MIDI controllers or instruments.

Moreover, the MIDI standard, while effective for encoding basic musical information, is limited in representing the complexities of Western music's time signatures and meters. It lacks an inherent framework to encode hierarchical metric structures, such as strong and weak beats, and struggles with the dynamic flexibility of metric changes. Additionally, its reliance on fixed temporal grids often oversimplifies expressive rhythmic nuances such as rubato, leading to a loss of critical musical details. These constraints necessitate supplementary metadata or advanced techniques to accurately capture the temporal intricacies of Western music.

Furthermore, a constraint emerges from the inadequate accessibility of ground truth data that clearly demarcates the differentiation between non-expressive and expressive MIDI tracks across all MIDI instruments for expressive performance detection. Presently, such data predominantly originate from piano and drum instruments in the GigaMIDI dataset.

7 CONCLUSION AND FUTURE WORK

Analyzing MIDI data may benefit symbolic music generation, computational musicology, and music data mining. The GigaMIDI dataset may contribute to MIR research by providing consolidated access to extensive MIDI

data for analysis. Metadata analyses, data source references, and findings on expressive music performance detection may enhance nuanced inquiries and foster progress in expressive music performance analysis and generation.

Our novel heuristics for discerning between non-expressive and expressively performed MIDI tracks exhibit notable efficacy in the presented dataset. The note onset median metric level (NOMML) heuristic demonstrates a classification accuracy of 100%, underscoring its discriminative capacity for expressive music performance detection.

Future work on the GigaMIDI dataset could significantly advance symbolic music research by using MIR techniques to identify and categorize musical styles systematically across all MIDI files. Currently, only about one-fifth of the dataset includes style metadata; expanding this would improve its comprehensiveness. Track-level, rather than file-level, style categorization would better capture the mix of styles in genres such as rock, jazz, and pop. Additionally, adding metadata for non-Western music, such as Asian classical or Latin/African styles, would reduce Western bias and offer a more inclusive resource for global music research, supporting cross-cultural studies.

8 DATA ACCESSIBILITY AND ETHICAL STATEMENTS

The GigaMIDI dataset consists of MIDI files acquired via the aggregation of previously available datasets and web scraping from publicly available online sources. Each subset is accompanied by source links, copyright information when available, and acknowledgments. File names are anonymized using MD5 hash encryption. We acknowledge the work from the previous dataset papers (Bosch et al., 2016; Callender et al., 2020; Choi et al., 2022; Crestel et al., 2018; Donahue et al., 2018; Ens and Pasquier, 2021; Foscarin et al., 2020; Gillick et al., 2019; Goebel, 1999; Hawthorne et al., 2019; Hu and Widmer, 2023; Hung et al., 2021; Hyun et al., 2022; Kong et al., 2022; Li et al., 2018; Liu et al., 2022; Ma et al., 2022; Miron et al., 2016; Müller et al., 2011; Plut et al., 2022; Raffel, 2016; Ryu et al., 2024; Sarmiento et al., 2021; Szelogowski et al., 2022; Wang et al., 2020; Zhang et al., 2022) that we aggregated and analyzed as part of the GigaMIDI subsets.

This dataset has been collected, utilized, and distributed under the Fair Dealing provisions for research and private study outlined in the Canadian Copyright Act (Government of Canada, 2024). Fair Dealing permits the limited use of copyright-protected material without the risk of infringement and without having to seek the permission of copyright owners. It is intended to provide a balance between the rights of creators and the rights of

users. As per instructions of the Copyright Office of Simon Fraser University,¹² two protective measures have been put in place that are deemed sufficient given the nature of the data (accessible online):

1. We explicitly state that this dataset has been collected, used, and distributed under the Fair Dealing provisions for research and private study outlined in the Canadian Copyright Act.
2. On the Hugging Face Hub, we advertise that the data are available for research purposes only and collect the user's legal name and email as proof of agreement before granting access.

We thus decline any responsibility for misuse.

The Findable, Accessible, Interoperable, Reusable (FAIR) principles (Jacobsen et al., 2020) serve as a framework to ensure that data are well-managed, easily discoverable, and usable for a broad range of purposes in research. These principles are particularly important in the context of data management to facilitate open science, collaboration, and reproducibility.

- **Findable:** Data should be easily discoverable by both humans and machines. This is typically achieved through proper metadata, traceable source links, and searchable resources. Applying this to MIDI data, each subset of MIDI files collected from public domain sources is accompanied by clear and consistent metadata via our GitHub and Hugging Face Hub webpages. For example, organizing the source links of each data subset, as done with the GigaMIDI dataset, ensures that each source can be easily traced and referenced, improving discoverability.
- **Accessible:** Once found, data should be easily retrievable using standard protocols. Accessibility does not necessarily imply open access, but it does mean that data should be available under well-defined conditions. For the GigaMIDI dataset, hosting the data on platforms such as Hugging Face Hub improves accessibility, as these platforms provide efficient data retrieval mechanisms, especially for large-scale datasets. Ensuring that MIDI data are accessible for public use while respecting any applicable licenses supports wider research and analysis in music computing.
- **Interoperable:** Data should be structured in such a way that it can be integrated with other datasets and used by various applications. MIDI data, being a widely accepted format in music research, are inherently interoperable, especially when standardized metadata and file formats are used. By ensuring that the GigaMIDI dataset complies with widely adopted standards and supports integration with state-of-the-art libraries in symbolic music processing, such as Symusic and MidiTok, the dataset

enhances its utility for music researchers and practitioners working across different platforms and systems.

- **Reusable:** Data should be well-documented and licensed to be reused in future research. Reusability is ensured through proper metadata, clear licenses, and documentation of provenance. In the case of GigaMIDI, aggregating all subsets from public domain sources and linking them to the original sources strengthens the reproducibility and traceability of the data. This practice allows future researchers to not only use the dataset but also verify and expand upon it by referring to the original data sources.

Developing ethical and responsible AI systems for music requires adherence to core principles of fairness, transparency, and accountability. The creation of the GigaMIDI dataset reflects a commitment to these values, emphasizing the promotion of ethical practices in data usage and accessibility. Our work aligns with prominent initiatives promoting ethical approaches to AI in music, such as AI for Music Initiatives,¹³ which advocates for principles guiding the ethical creation of music with AI, supported by the Metacreation Lab for Creative AI¹⁴ and the Centre for Digital Music,¹⁵ which provide critical guidelines for the responsible development and deployment of AI systems in music. Similarly, the Fairly Trained initiative¹⁶ highlights the importance of ethical standards in data curation and model training, principles that are integral to the design of the GigaMIDI dataset. These frameworks have shaped the methodologies used in this study, from dataset creation and validation to algorithmic design and system evaluation. By engaging with these initiatives, this research not only contributes to advancing AI in music but also reinforces the ethical use of data for the benefit of the broader music computing and MIR communities.

ACKNOWLEDGEMENTS

We gratefully acknowledge the support and contributions that have directly or indirectly aided this research. This work was supported in part by funding from the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Social Sciences and Humanities Research Council of Canada (SSHRC). We also extend our gratitude to the School of Interactive Arts and Technology (SIAT) at Simon Fraser University (SFU) for providing resources and an enriching research environment. Additionally, we thank the Centre for Digital Music (C4DM) at Queen Mary University of London (QMUL) for fostering collaborative opportunities and supporting our engagement with interdisciplinary research initiatives. We also acknowledge the support of Engineering and Physical Sciences Research Council (EPSRC) UK Research and Innovation (UKRI) Centre for Doctoral Training in AI and Music

(Grant EP/S022694/1) and UKRI - Innovate UK (Project number 10102804).

Special thanks are extended to Dr. Cale Plut for his meticulous manual curation of musical styles and to Dr. Nathan Fradet for his invaluable assistance in developing the Hugging Face Hub website for the GigaMIDI dataset, ensuring it is accessible and user-friendly for music computing and MIR researchers. We also sincerely thank our research interns, Paul Triana and Davide Rizzotti, for their thorough proofreading of the manuscript, as well as the TISMIR reviewers who helped us improve our manuscript.

Finally, we express our heartfelt appreciation to the individuals and communities who generously shared their MIDI files for research purposes. Their contributions have been instrumental in advancing this work and fostering collaborative knowledge in the field.

COMPETING INTERESTS

The authors have no competing interests to declare.

AUTHORS' CONTRIBUTIONS

The authors confirm their contributions to the manuscript as follows:

- **Study Conception and Design:** Keon Ju Maverick Lee, Jeff Ens, Pedro Sarmiento, Mathieu Barthet, and Philippe Pasquier.
- **Data Collection and Metadata:** Keon Ju Maverick Lee, Jeff Ens, Pedro Sarmiento, and Sara Adkins.
- **Expressive Music Performance Heuristic Design and Experimentation:** Keon Ju Maverick Lee, and Jeff Ens.
- **Analysis and Interpretation of Results:** Keon Ju Maverick Lee, Jeff Ens, Pedro Sarmiento, Mathieu Barthet, and Philippe Pasquier.
- **Manuscript Draft Preparation:** Keon Ju Maverick Lee, Jeff Ens, Sara Adkins, and Pedro Sarmiento.
- **Research Guidance and Advisement:** Philippe Pasquier and Mathieu Barthet.

All authors have reviewed the results and approved the final version of the manuscript.

NOTES

1. <https://zenodo.org/>
2. <https://github.com/>
3. <https://github.com/Metacreation-Lab/GigaMIDI-Dataset/tree/main>
4. <https://github.com/Yikai-Liao/symusic>
5. <https://github.com/Natooz/MidiTok>
6. <https://huggingface.co/datasets/Metacreation/GigaMIDI>
7. <https://github.com/Metacreation-Lab/MetaMIDI-Dataset>
8. <https://www.discogs.com/>

9. <https://www.last.fm/>
10. <http://www.tagtraum-music.com/>
11. <https://github.com/metabrainz/musicbrainz-server>
12. <https://www.lib.sfu.ca/help/academic-integrity/copyright>
13. <https://aiformusic.info/>
14. <http://metacreation.net/>
15. <https://www.c4dm.eecs.qmul.ac.uk/>
16. <https://www.fairlytrained.org/>

ADDITIONAL FILES

The additional files for this article can be found using the links below:

- **Supplementary Appendix A.** DOI: <https://doi.org/10.5334/tismir.203.s1>.
- **Supplementary Appendix B.** DOI: <https://doi.org/10.5334/tismir.203.s2>.

AUTHOR AFFILIATIONS

Keon Ju Maverick Lee  <https://orcid.org/0009-0007-3324-8992>

Metacreation Lab for Creative AI, School of Interactive Arts and Technology, Simon Fraser University, Vancouver, Canada

Jeff Ens

Metacreation Lab for Creative AI, School of Interactive Arts and Technology, Simon Fraser University, Vancouver, Canada

Sara Adkins

Independent Researcher, Boston, United States of America

Pedro Sarmiento

Centre for Digital Music, Queen Mary University of London, London, United Kingdom

Mathieu Barthet

Centre for Digital Music, Queen Mary University of London, London, United Kingdom, and Aix-Marseille Univ CNRS PRISM, Marseille, France

Philippe Pasquier

Metacreation Lab for Creative AI, School of Interactive Arts and Technology, Simon Fraser University, Vancouver, Canada

REFERENCES

- Adkins, S., Sarmiento, P., and Barthet, M.** (2023, April). LooperGP: A loopable sequence model for live coding performance using GuitarPro tablature. In *Proceedings of International Conference on Computational Intelligence in Music, Sound, Art and Design (Part of EvoStar)*, Brno, Czech Republic.
- Arcos, J. L., de Mántaras, R. L., and Serra, X.** (1998). Saxex: A case-based reasoning system for generating expressive musical performances*. *Journal of New Music Research*, 27(3), 194–210. <https://doi.org/10.1080/09298219808570746>
- Barthet, M., Depalle, P., Kronland-Martinet, R., and Ystad, S.** (2010). Acoustical correlates of timbre and expressiveness

- in clarinet performance. *Music Perception*, 28(2), 135–154. <https://doi.org/10.1525/mp.2010.28.2.135>
- Bellini, P., Bruno, I., and Nesi, P.** (2005). Automatic formatting of music sheets through MILLA rule-based language and engine. *Journal of New Music Research*, 34(3), 237–257. <https://doi.org/10.1080/09298210500236051>
- Berndt, A., and Hähnel, T.** (2010, September). Modelling musical dynamics. In *Proceedings of the Audio Mostly Conference on Interaction with Sound*, Piteå, Sweden.
- Born, G.** (2020). Diversifying MIR: Knowledge and real-world challenges, and new interdisciplinary futures. *Transactions of the International Society for Music Information Retrieval*, 3(1), 193–204. <https://doi.org/10.5334/tismir.58>
- Bosch, J. J., Marxer, R., and Gómez, E.** (2016). Evaluation and combination of pitch estimation methods for melody extraction in symphonic classical music. *Journal of New Music Research*, 45(2), 101–117. <https://doi.org/10.1080/09298215.2016.1182191>
- Boulanger-Lewandowski, N., Bengio, Y., and Vincent, P.** (2012, June). Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. In *Proceedings of the International Conference on Machine Learning*, Edinburgh, Scotland.
- Briot, J.-P.** (2021). From artificial neural networks to deep learning for music generation: history, concepts and trends. *Journal of Neural Computing and Applications*, 33, 39–65. <https://doi.org/10.1007/s00521-020-05399-0>
- Briot, J.-P., and Pachet, F.** (2020). Deep learning for music generation: Challenges and directions. *Journal of Neural Computing and Applications*, 32, 981–993. <https://doi.org/10.1007/s00521-018-3813-6>
- Brunner, G., Konrad, A., Wang, Y., and Wattenhofer, R.** (2018, September). MIDI-VAE: Modeling dynamics and instrumentation of music with applications to style transfer. In *Proceedings of the International Society for Music Information Retrieval Conference*, Paris, France.
- Callender, L., Hawthorne, C., and Engel, J.** (2020). Improving perceptual quality of drum transcription with the expanded Groove MIDI dataset. <https://magenta.tensorflow.org/oaf-drums>
- Cancino-Chacón, C. E., Grachten, M., Goebel, W., and Widmer, G.** (2018). Computational models of expressive music performance: A comprehensive and critical review. *Journal of Frontiers in Digital Humanities*, 5, Article 25. <https://doi.org/10.3389/fdigh.2018.00025>
- Cancino-Chacón, C., Peter, S. D., Karystinaios, E., Foscari, F., Grachten, M., and Widmer, G.** (2022). Partitura: A Python package for symbolic music processing. In *Proceedings of the Music Encoding Conference*, Halifax, Canada.
- Choi, E., Chung, Y., Lee, S., Jeon, J., Kwon, T., and Nam, J.** (2022, December). YM2413-MDB: A multi-instrumental fm video game music dataset with emotion annotations. In *Proceedings of the International Society for Music Information Retrieval Conference*, Bengaluru, India.
- Collins, T., and Barthet, M.** (2023, November). Expressor: A transformer model for expressive MIDI performance. In *Proceedings of the 16th International Symposium on Computer Music Multidisciplinary Research (CMMR)*, Tokyo, Japan.
- Crauwels, K.** (2016). Musicmap. <https://musicmap.info/>
- Crawford, T., and Lewis, R.** (2016). Review: Music encoding initiative. *Journal of the American Musicological Society*, 69(1), 273–285. <https://doi.org/10.1525/jams.2016.69.1.273>
- Crestel, L., Esling, P., Heng, L., and McAdams, S.** (2018, September). A database linking piano and orchestral MIDI scores with application to automatic projective orchestration. In *Proceedings of the International Society for Music Information Retrieval Conference*, Paris, France.
- Cros Vila, L., and Sturm, B. L. T.** (2023, August). Statistical evaluation of ABC-formatted music at the levels of items and corpora. In *Proceedings of AI Music Creativity (AIMC)*, Brighton, UK.
- Cuthbert, M. S., and Ariza, C.** (2010, August). music21: A toolkit for computer-aided musicology and symbolic music data. In *Proceedings of the International Society for Music Information Retrieval Conference*, Utrecht, Netherlands.
- Dannenberg, R. B.** (2006, November). The interpretation of MIDI velocity. In *Proceedings of the International Computer Music Conference*, New Orleans, United States.
- Dieleman, S., Van Den Oord, A., and Simonyan, K.** (2018, December). The challenge of realistic music generation: Modelling raw audio at scale. In *Proceedings of Conference on Neural Information Processing Systems*, Montreal, Canada.
- Donahue, C., Mao, H. H., and McAuley, J.** (2018, September). The NES music database: A multi-instrumental dataset with expressive performance attributes. In *Proceedings of the International Society for Music Information Retrieval Conference*, Paris, France.
- Edwards, D., Dixon, S., Benetos, E., Maezawa, A., and Kusaka, Y.** (2024). A data-driven analysis of robust automatic piano transcription. *IEEE Signal Processing Letters*, 31, 681–685. <https://doi.org/10.1109/LSP.2024.3363646>
- Ens, J., and Pasquier, P.** (2020, October). MMM: Exploring conditional multi-track music generation with the transformer. In *Proceedings of the International Society for Music Information Retrieval Conference*, Montreal, Canada.
- Ens, J., and Pasquier, P.** (2021, November). Building the MetaMIDI dataset: Linking symbolic and audio musical data. In *Proceedings of the International Society for Music Information Retrieval Conference*, Online.
- Foscari, F., McLeod, A., Rigaux, P., Jacquemard, F., and Sakai, M.** (2020, October). ASAP: A dataset of aligned scores and performances for piano transcription. In *Proceedings of the International Society for Music Information Retrieval Conference*, Montreal, Canada.

- Fradet, N., Briot, J.-P., Chhel, F., Seghrouchni, A. E. F., and Gutowski, N.** (2021, November). MidiTok: A python package for MIDI file tokenization. In *Proceedings of the International Society for Music Information Retrieval Conference, Online*.
- Gillick, J., Roberts, A., Engel, J., Eck, D., and Bamman, D.** (2019, June). Learning to groove with inverse sequence transformations. In *Proceedings of the International Conference on Machine Learning*, Long Beach, United States.
- Goddard, C., Barthet, M., and Wiggins, G.** (2018). Assessing musical similarity for computational music creativity. *Journal of the Audio Engineering Society*, 66(4), 267–276. <https://doi.org/10.17743/jaes.2018.0012>
- Goebel, W.** (1999). The Vienna 4 × 22 Piano Corpus. https://repo.mdw.ac.at/projects/IWK/the_vienna_4x22_piano_corpus/index.html
- Goebel, W.** (2001). Melody lead in piano performance: Expressive device or artifact? *Journal of the Acoustical Society of America*, 110(1), 563–572. <https://doi.org/10.1121/1.1376133>
- Gómez-Marín, D., Jordà, S., and Herrera, P.** (2020). Drum rhythm spaces: From polyphonic similarity to generative maps. *Journal of New Music Research*, 49(5), 438–456. <https://doi.org/10.1080/09298215.2020.1806887>
- Government of Canada. (2024). The Canadian Copyright Act, RSC 1985, c. C-42, s. 29 (fair dealing for research and private study). <https://laws-lois.justice.gc.ca/eng/acts/C-42/section-29.html>
- Grachten, M., and Widmer, G.** (2012). Linear basis models for prediction and analysis of musical expression. *Journal of New Music Research*, 41(4), 311–322. <https://doi.org/10.1080/09298215.2012.731071>
- Hawthorne, C., Stasyuk, A., Roberts, A., Simon, I., Huang, C.-Z. A., Dieleman, S., Elsen, E., Engel, J., and Eck, D.** (2019, May). Enabling factorized piano music modeling and generation with the MAESTRO dataset. In *Proceedings of the International Conference on Learning Representations (ICLR)*, New Orleans, United States.
- Hernandez-Olivan, C., and Beltrán, J. R.** (2022). Music composition with deep learning: A review. In **A. Biswas, E. Wenekes, A. Wiczorkowska, and R. H. Laskar** (Eds.), *Advances in speech and music technology: computational aspects and applications*, (pp. 25–50). Springer. <https://doi.org/10.1007/978-3-031-18444-4>
- Hu, P., and Widmer, G.** (2023, November). The Batik-plays-Mozart Corpus: Linking performance to score to musicological annotations. In *Proceedings of the International Society for Music Information Retrieval Conference*, Milan, Italy.
- Huang, C.-Z. A., Vaswani, A., Uszkoreit, J., Shazeer, N., Simon, I., Hawthorne, C., Dai, A. M., Hoffman, M. D., Dinculescu, M., and Eck, D.** (2019, May). Music transformer. In *Proceedings of the International Conference on Learning Representations (ICLR)*, New Orleans, United States.
- Hung, H.-T., Ching, J., Doh, S., Kim, N., Nam, J., and Yang, Y.-H.** (2021, November). EMOPIA: A multi-modal pop piano dataset for emotion recognition and emotion-based music generation. In *Proceedings of the International Society for Music Information Retrieval Conference, Online*.
- Hyun, L., Kim, T., Kang, H., Ki, M., Hwang, H., Park, K., Han, S., and Kim, S. J.** (2022, December). ComMU: Dataset for combinatorial music generation. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, New Orleans, United States.
- Jacobsen, A., de Miranda Azevedo, R., Juty, N., Batista, D., Coles, S., Cornet, R., Courtot, M., Crosas, M., Dumontier, M., Evelo, C. T., Goble, C., Guizzardi, G., Hansen, K. K., Hasnain, A., Hettne, K., Heringa, J., Hooft, R. W., Imming, M., Jeffery, K. G., and Schultes, E.** (2020). FAIR principles: Interpretations and implementation considerations. *Data Intelligence*, 2(1-2), 10–29. https://doi.org/10.1162/dint_r_00024
- Kim, H., Miron, M., and Serra, X.** (2022, December). Note level MIDI velocity estimation for piano performance. In *Proceedings of the International Society for Music Information Retrieval Conference*, Bengaluru, India.
- Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M., and Klein, M.** (2002). *Logistic regression*. Springer. <https://doi.org/10.1007/978-1-4419-1742-3>
- Kong, Q., Li, B., Chen, J., and Wang, Y.** (2022). GiantMIDI-Piano: A large-scale MIDI dataset for classical piano music. *Transactions of the International Society for Music Information Retrieval*. <https://doi.org/10.5334/tismir.80>
- Kuo, C.-S., Chen, W.-K., Liu, C.-H., and You, S. D.** (2021, September). Velocity prediction for MIDI notes with deep learning. In *Proceedings of IEEE International Conference on Consumer Electronics-Taiwan*, Penghu, Taiwan.
- Li, B., Liu, X., Dinesh, K., Duan, Z., and Sharma, G.** (2018). Creating a multitrack classical music performance dataset for multimodal music analysis: Challenges, insights, and applications. *IEEE Transactions on Multimedia*, 21(2), 522–535. <https://doi.org/10.48550/arXiv.1612.08727>
- Li, T., Ogihara, M., and Tzanetakis, G.** (2012). *Music data mining*. CRC Press. <https://doi.org/10.1201/b11041>
- Licata, T.** (2002). *Electroacoustic music: Analytical perspectives*. Bloomsbury Publishing USA. ISBN-10: 0313314209.
- Liu, J., Dong, Y., Cheng, Z., Zhang, X., Li, X., Yu, F., and Sun, M.** (2022, December). Symphony generation with permutation invariant language model. In *Proceedings of the International Society for Music Information Retrieval Conference*, Bengaluru, India.
- Ma, X., Liu, X., Zhang, B., and Wang, Y.** (2022, December). Robust melody track identification in symbolic music. In *Proceedings of International Society for Music Information Retrieval Conference*, Bengaluru, India.
- Manzelli, R., Thakkar, V., Siahkamari, A., and Kulis, B.** (2018, September). Conditioning deep generative raw audio models for structured automatic music. In *Proceedings of International Society for Music Information Retrieval Conference*, Paris, France.

- Mauch, M., MacCallum, R. M., Levy, M., and Leroi, A. M.** (2015). The evolution of popular music: USA 1960–2010. *Royal Society Open Science*, 2(5), 150081. <https://doi.org/10.1098/rsos.150081>
- Meroño-Peñuela, A., Hoekstra, R., Gangemi, A., Bloem, P., de Valk, R., Stringer, B., Janssen, B., de Boer, V., Allik, A., and Schlobach, S.** (2017). The MIDI linked data cloud. In *The Semantic Web–ISWC 2017:16th International Semantic Web Conference, Vienna, Austria, October 21–25, 2017, Proceedings, Part II 16* (Vol. 16, pp. 156–164). Springer. https://doi.org/10.1007/978-3-319-68204-4_16
- MIDI Association. (1996a). The complete MIDI 1.0 detailed specification. <https://midi.org/spec-detail>
- MIDI Association. (1996b). General MIDI instrument group and mapping. <https://midi.org/general-midi-level-1>
- Miron, M., Carabias-Orti, J. J., Bosch, J. J., Gómez, E., Janer, J., et al.** (2016). Score-informed source separation for multichannel orchestral recordings. *Journal of Electrical and Computer Engineering*. <https://doi.org/10.1155/2016/8363507>
- Müller, M., Konz, V., Bogler, W., and Arifi-Müller, V.** (2011, October). Saarland music data (SMD). In *Proceedings of the international society for music information retrieval conference (ISMIR): late breaking session*.
- Ortega, F. J., Giraldo, S. I., Perez, A., and Ramírez, R.** (2019). Phrase-level modelling of expression in violin performances. *Journal of Frontiers in Psychology*, 10, 776. <https://doi.org/10.3389/fpsyg.2019.00776>
- Palmer, C.** (1997). Music performance. *Annual Review of Psychology*, 48(1), 115–138. <https://doi.org/10.1146/annurev.psych.48.1.115>
- Payne, C.** (2019). MuseNet (OpenAI). <https://openai.com/blog/musenet>
- Plut, C., Pasquier, P., Ens, J., and Tchemeube, R.** (2022). The IsoVAT corpus: parameterization of musical features for affective composition. *Transactions of the International Society for Music Information Retrieval (TISMIR)*, 5(1), 173–189. <https://doi.org/10.5334/tismir.120>
- Qiu, L., Li, S., and Sung, Y.** (2021). DBTMPE: Deep bidirectional transformers-based masked predictive encoder approach for music genre classification. *Mathematics*, 9(5), 530. <https://doi.org/10.3390/math9050530>
- Raffel, C.** (2016). The Lakh MIDI dataset v0.1. <https://colinraffel.com/projects/lmd/>
- Raffel, C., and Ellis, D. P.** (2016a, March). Optimizing DTW-based audio-to-MIDI alignment and matching. In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, Shanghai, China.
- Raffel, C., and Ellis, D. P. W.** (August, 2016b). Extracting ground truth information from MIDI files: A MIDIffesto. In *Proceedings of the International Society for Music Information Retrieval Conference*, New York, United States.
- Repp, B. H.** (1997a). Acoustics, perception, and production of legato articulation on a computer-controlled grand piano. *Journal of the Acoustical Society of America*, 102(3), 1878–1890. <https://doi.org/10.1121/1.420110>
- Repp, B. H.** (1997b). The aesthetic quality of a quantitatively average music performance: Two preliminary experiments. *Music Perception*, 14(4), 419–444. <https://doi.org/10.2307/40285732>
- Ryu, J., Rhyu, S., Yoon, H.-G., Kim, E., Yang, J. Y., and Kim, T.** (2024, February). MID-FILD: MIDI dataset for fine-level dynamics. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vancouver, Canada.
- Sarmiento, P., Kumar, A., Carr, C., Zukowski, Z., Barthet, M., and Yang, Y.-H.** (2021, November). DadaGP: A dataset of tokenized GuitarPro songs for sequence models. In *Proceedings of International Society for Music Information Retrieval Conference, Online*.
- Sarmiento, P., Kumar, A., Chen, Y.-H., Carr, C., Zukowski, Z., and Barthet, M.** (2023a, April). GTR-CTRL: Instrument and genre conditioning for guitar-focused music generation with transformers. In *Proceedings of the EvoMUSART Conference*, Brno, Czech Republic.
- Sarmiento, P., Kumar, A., Xie, D., Carr, C., Zukowski, Z., and Barthet, M.** (2023b, November). ShredGP: Guitarist style-conditioned tablature generation. In *The 16th International Symposium on Computer Music Multidisciplinary Research*, Tokyo, Japan.
- Schaffrath, H.** (1995). The Essen Folksong Collection. <http://www.esac-data.org/>
- Shih, Y.-J., Wu, S.-L., Zalkow, F., Muller, M., and Yang, Y.-H.** (2022). Theme Transformer: symbolic music generation with theme-conditioned transformer. *IEEE Transactions on Multimedia*. <https://doi.org/10.48550/arXiv.2111.04093>
- Simonetta, F., Carnovalini, F., Orio, N., and Rodà, A.** (2018, September). Symbolic music similarity through a graph-based representation. In *Proceedings of the Audio Mostly Conference*, Wrexham, United Kingdom.
- Sitarz, M.** (2022). Extending F1 metric, probabilistic approach. *Journal of Advances in Artificial Intelligence and Machine Learning*, 3, 1025–1038. <https://doi.org/10.48550/arXiv.2210.11997>
- Szelogowski, D., Mukherjee, L., and Whitcomb, B.** (2022, December). A novel dataset and deep learning benchmark for classical music form recognition and analysis. In *Proceedings of International Society for Music Information Retrieval Conference*, Bengaluru, India.
- Tang, J., Wiggins, G., and Fazekas, G.** (2023, November). Reconstructing human expressiveness in piano performances with a Transformer network. In *Proceedings of International Symposium on Computer Music Multidisciplinary Research (CMMR)*, Tokyo, Japan.
- vo Rütte, D., Biggio, L., Kilcher, Y., and Hofmann, T.** (2023). FIGARO: generating symbolic music with fine-grained artistic control. In *Proceedings of International Conference on Learning Representations*, Kigali, Rwanda.
- Wang, Z., Chen, K., Jiang, J., Zhang, Y., Xu, M., Dai, S., Gu, X., and Xia, G.** (2020, October). POP909: A pop-song dataset for music arrangement generation. In *Proceedings of*

International Society for Music Information Retrieval Conference, Montreal, Canada.

Wong, T.-T. (2015). Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation. *Pattern Recognition*, 48(9), 2839–2846. <https://doi.org/10.1016/j.patcog.2015.03.009>

Zeng, M., Tan, X., Wang, R., Ju, Z., Qin, T., and Liu, T.-Y. (2021, August). Musicbert: Symbolic music understanding with large-scale pre-training. In *Proceedings of the Joint Conference of the Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing*, Bangkok, Thailand.

Zhang, H., Tang, J., Rafee, S. R. M., and Fazekas, S. D. G. (2022, December). ATEPP: A dataset of automatically transcribed expressive piano performance. In *Proceedings of International Society for Music Information Retrieval Conference*, Bengaluru, India.

Zhang, N. (2020). Learning adversarial transformer for symbolic music generation. *IEEE Transactions on Neural Networks and Learning Systems*. <https://doi.org/10.1109/TNNLS.2020.2990746>

Zukowski, Z., and Carr, C. (2017, December). Generating black metal and math rock: Beyond Bach, Beethoven, and Beatles. In *NIPS Workshop on Machine Learning for Creativity and Design*, Long Beach, United States.

TO CITE THIS ARTICLE:

Lee, K.J.M., Ens, J., Adkins, S., Sarmiento, P., Barthet, M., & Pasquier, P. (2025). The GigaMIDI Dataset with Features for Expressive Music Performance Detection. *Transactions of the International Society for Music Information Retrieval*, 8(1), 1–19. DOI: <https://doi.org/10.5334/tismir.203>

Submitted: 14 May 2024 **Accepted:** 11 November 2024 **Published:** 7 February 2025

COPYRIGHT:

© 2025 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <https://creativecommons.org/licenses/by/4.0/>.

Transactions of the International Society for Music Information Retrieval is a peer-reviewed open access journal published by Ubiquity Press.