

MIDICAPS: A LARGE-SCALE MIDI DATASET WITH TEXT CAPTIONS

Jan Melechovsky *, Abhinaba Roy *, Dorien Herremans

Singapore University of Technology and Design

jan_melechovsky@mymail.sutd.edu.sg, abhinaba_roy@sutd.edu.sg, dorien_herremans@sutd.edu.sg

ABSTRACT

Generative models guided by text prompts are increasingly becoming more popular. However, no text-to-MIDI models currently exist due to the lack of a captioned MIDI dataset. This work aims to enable research that combines LLMs with symbolic music by presenting **MidiCaps**, the first openly available large-scale MIDI dataset with text captions. MIDI (Musical Instrument Digital Interface) files are widely used for encoding musical information and can capture the nuances of musical composition. They are widely used by music producers, composers, musicologists, and performers alike. Inspired by recent advancements in captioning techniques, we present a curated dataset of over 168k MIDI files with textual descriptions. Each MIDI caption describes the musical content, including tempo, chord progression, time signature, instruments, genre, and mood, thus facilitating multi-modal exploration and analysis. The dataset encompasses various genres, styles, and complexities, offering a rich data source for training and evaluating models for tasks such as music information retrieval, music understanding, and cross-modal translation. We provide detailed statistics about the dataset and have assessed the quality of the captions in an extensive listening study. We anticipate that this resource will stimulate further research at the intersection of music and natural language processing, fostering advancements in both fields.

1. INTRODUCTION

The recent development of large-language models (LLMs) has revolutionised how we interact with text, images, and even audio. By incorporating elements of multimodal learning, researchers have combined LLMs with other modalities. The resulting models can analyze and generate accurate descriptions and captions, which in turn facilitates downstream tasks such as question answering [1], image generation [2], and music generation [3]. However, we have yet to see such an evolution for MIDI files.

*These authors contributed equally to this work.

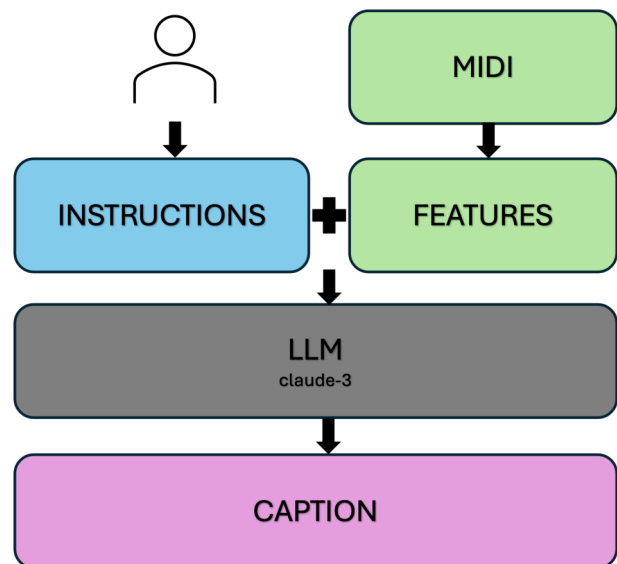


Figure 1: Overview of our approach. We extract meaningful and relevant features from MIDI files. These features are then added as tags to the human instructions that are sent to an LLM (Claude-3) to generate meaningful text captions of MIDI files.

In the field of Music Information Retrieval (MIR), MIDI plays a crucial role as a symbolic and musically meaningful representation of music. The format is often used by music producers and composers working in Digital Audio Workstations (DAWs). It is also a useful format for the computational analysis of music and related tasks such as music transcription, genre classification, similarity measurement, and music recommendation [4]. Furthermore, due to the symbolic nature of music, it has long been used by music generation algorithms [5]. In recent years, we have seen a surge of interest in music generation from free-flow text instructions [3, 6–9]. These studies leverage the expressive capabilities of LLMs to translate textual representation of musical attributes into actual music audio. This necessitates a meticulous alignment between the textual and musical feature spaces to ensure that the generated music closely follows text instructions. To validate and establish benchmarks for this text-to-music mapping, large-scale datasets with text captions have been developed [3, 10].

No such efforts, however, have yet been made for the MIDI format, despite its widespread use by musicians and its obvious, historically supported usage in music generation. This lack of text-MIDI datasets, in turn, has inhibited

researchers from exploring interesting and novel tasks such as MIDI generation from free-flow text prompts. In this work, we identify this shortcoming and develop a robust solution in the form of a large-scale curated MIDI dataset accompanied by text captions. Our goal is to obtain captions that are i) large in volume, ii) contain accurate information about the musical contents, and iii) feature a rich and refined vocabulary. We posit that such a dataset-level approach opens up further opportunities for researchers in MIDI-LLMs-related tasks.

To address the first goal, we identify an open source large-scale MIDI dataset in the form of Lakh MIDI dataset [11], that contains over 170K MIDI examples. Second, to attain the musical contents in each MIDI file, we extract meaningful features encompassing tempo, chord progression, time signature, instruments present, genre and mood. Each of the features are extracted using state-of-the-art MIR tools that ensure the quality and accuracy of the features extracted. After feature extraction, we are still left with the task of caption generation. Relying on traditional human annotation is tedious, time-consuming, and costly. Instead, motivated by the recent success of LLMs, we utilize in-context learning – a model’s ability to temporarily learn from human-provided instructions [12]. Our decision is motivated by Melechovsky et al. [3], who have demonstrated the efficacy of in-context learning in generating captions that are accurate, rich in description as well as grammatically coherent. In our approach, we furnish the LLM with instructions to generate captions based on the extracted music features, supplemented by a small set of feature-caption pairs created by expert annotators. Given the current absence of freely available MIDI-caption datasets, we anticipate that the provision of a substantial volume of detailed and informative captions will inspire the research community to delve further into tasks related to MIDI and Large Language Models (LLMs). The main contributions of this work can be summarized as follows:

- We introduce the first curated large-scale open dataset of MIDI-caption pairs, termed **MidiCaps**¹.
- Furthermore, we present a comprehensive set of music-specific features extracted from MIDI files. These features succinctly characterize the musical content, encompassing tempo, chord progression, time signature, instrument presence, genre, and mood.
- Finally, we provide a text caption annotation framework tailored specifically for MIDI data (see Figure 1). Leveraging the in-context learning capability of large language models (LLMs), we enable the generation of captions using only a small number of feature-caption training pairs. This framework, a first of its kind, is made freely accessible to users², facilitating the generation of MIDI-caption pairs for their individual MIDI files.

2. RELATED WORK

To the best of our knowledge, there are no publicly available MIDI caption datasets. In this section, we briefly mention various publicly available MIDI datasets and discuss the closely related topic of caption generation from audio and music.

Despite the scarcity of MIDI caption datasets, existing repositories offer potential resources that could be adapted for this purpose. Among these, the Lakh MIDI Dataset [11] stands out, comprising a vast collection of MIDI files. While primarily tailored for MIR tasks such as melody extraction and chord estimation, its volume and diversity present an opportunity for repurposing towards captioning tasks, albeit requiring appropriate preprocessing. The MAESTRO Dataset [13] offers aligned pairs of MIDI and audio files, primarily for piano music generation. The MuseGAN Dataset [14] focuses on multi-track songs, and the MAPS Dataset [15], contains recordings of classical piano pieces alongside aligned MIDI files and thus also present potential avenues for MIDI captioning research. Additionally, the Wikifonia Dataset [16] features a substantial collection of lead sheets accompanied by MIDI files. Closest to our proposed MIDI-caption dataset is the WikiMusicText (WikiMT) dataset [17], which includes lead sheets in ABC notation with metadata including text descriptions. These descriptions, however, pertain to general information about the music piece rather than detailed descriptions of musical contents provided in MIDI files within our captions.

In the last three years, several models were released for automatic caption generation from music audio files. One of the earlier models, MusCaps [18], uses an architecture based on recurrent and convolutional layers as well as a multimodal encoder. Recent research includes the use of large language models (LLMs) for captioning [3, 7, 10]. In [7], a pseudo labeling approach is used to label a large training dataset. First, existing captions from other datasets are curated, then the MuLaN [19] model, a joint music-text embedding model, evaluates the distance between captions and unlabeled audio files. The top caption candidates are selected based on their frequency to ensure balance among all samples. In [20], the focus is on capturing the full sentiment of classical music recordings through text descriptions, introducing a Group-Topology Preservation Loss to be used with their cross-modality translation model. A recent study by Doh et al. [10] targets pseudo labeling of audio data with the help of an LLM, utilizing the MusicCaps [6] dataset as ground truth and instructing GPT-3.5 Turbo [21] to generate full captions from these tags.

In [3], Melechovsky et al. curate a new dataset based on the MusicCaps dataset [6], called MusicBench. In MusicBench, the original captions are enhanced by including additional music descriptors such as chord sequence, musical key, time signature, and tempo. After performing audio and text augmentations to expand the dataset size, they use ChatGPT [22] for rephrasing captions to create more diverse captions. Furthermore, they employ in-context learning to guide ChatGPT using a small set

¹huggingface.co/datasets/amaai-lab/MidiCaps

²github.com/AMAAI-Lab/MidiCaps

of human-annotated examples, instructing it to generate diverse captions to create an evaluation dataset from extracted tags, named FMACaps. Inspired by their methodology, we adopt a similar approach and utilize in-context learning alongside a large-language model to generate captions from MIDI features. In the subsequent section, we offer an in-depth description of our proposed framework for MIDI captioning.

3. METHOD

In this section, we discuss details regarding the music-specific features we extract from MIDI files.

3.1 Feature extraction

In a first step, as per Figure 2, we extract various musical features from the MIDI files. This is achieved in two ways: a number of features are extracted directly from the MIDI files, and others are extracted from the synthesized MIDI files. The details of our approach are described below.

3.1.1 Preprocessing

We preprocess all files to remove faulty files. For instance, we found multiple files that had never-ending notes. Using Mido [23], we further exclude files of duration shorter than 3 seconds and longer than 15 minutes.

3.1.2 MIDI feature extraction

We use Music21 [24] and Mido [23] libraries to extract the following features from MIDI: Musical Key (Music21), Time Signature (Music21), Tempo (Mido), Duration of the MIDI file (Mido), and a list of Instruments (Mido).

The **Key** and **Time Signature** features are obtained through `music21.midi.analyze('keys')` and `music21.midi.getTimeSignatures()` functions, respectively. To calculate the **Tempo**, we first look for the `set_tempo` MIDI message to get the MIDI tempo. Then, the `mido.tempo2bpm()` function is used to convert this MIDI tempo to beats per minute (bpm). For MIDI file **Duration**, we retrieve the `length` attribute of a `mido.MidiFile` object.

To extract a list of **instruments**, we filter the MIDI messages based on channel number and their associated instrument program obtained from the program change message. To treat ambiguity given by some faulty files, we always take the last assigned program number as the definite instrument number for each MIDI channel. For channel 10, which is reserved for drums, we always consider the assigned instrument to be drums, unless there is another percussion instrument specified.

We further process the extracted instruments in three steps to identify the most prominent instruments. First, we extract total note duration for each of the instruments by scraping through note-on and note-off messages, and rank them by this duration. Second, we map the program numbers to their respective instrument names, grouping similar variations (e.g., both nylon and steel string acoustic guitars

as ‘acoustic guitar’). Third, we reduce the list of instruments to only include one instance of the same instrument name (in the previous example, the two acoustic guitars would merge into one), and then take top five instruments sorted by their total note duration.

3.1.3 Synthesized audio feature extraction

We use the Midi2Audio library [25] that utilizes FluidSynth [26, 27] to synthesize audio from MIDI with the Fluid Release 3 General-MIDI sound font. Then, we use these audio files to extract genre, mood, and chord features.

To extract **genre** and **mood**, we use Essentia [28], specifically the MTG-Jamendo genre and mood/theme discogs effnet models³. We keep the top two genre tags with the highest confidence score, and the top five mood/theme tags, also based on their confidence score. The confidence scores for each tag are also stored.

Next, we extract the single most occurring **chord sequence** of length 3 to 5. To obtain this, we first extract all chords from the audio using Chordino [29]. To obtain the most frequent short chord sequence, we first iterate through the chord list to find the most frequent patterns consisting of 3, 4, and 5 consequent chords. We do not allow these patterns to have the same first and last chord, e.g., [A, B, C, A] for a pattern of length 4 is not allowed, as this is likely an [A, B, C] pattern of length 3. Then, we decide on which pattern to keep through a set of rules described in Algorithm 1. In the below algorithm n_i represents the occurrence count of the most frequent pattern (p_i) of length i . We save the final selected pattern along with a number denoting how many times it occurred. Once we have extracted all of the features extracted, we move on to caption generation, described in the next subsection.

Algorithm 1 Selecting the most frequent chord pattern.

```

▷  $p_i$ : most frequent pattern of length  $i$ 
▷  $n_i$ : occurrence count of  $p_i$ 
▷  $p$ : final selected most frequent pattern
 $n = n_3 + n_4 + n_5$ 
if ( $n_5 \geq 0.8 \cdot n_4$ ) & ( $n_5 \geq 0.25 \cdot n$ ) then
     $p \leftarrow p_5$ 
else if ( $n_4 \geq 0.8 \cdot n_3$ ) & ( $n_4 \geq 0.3 \cdot n$ ) then
     $p \leftarrow p_4$ 
else if ( $n_3 == 0$ ) then
    if ( $n_4 == 0$ ) then
        if ( $n_5 == 0$ ) then
             $p \leftarrow \text{None}$ 
        else
             $p \leftarrow p_5$ 
        end if
    else
         $p \leftarrow p_4$ 
    end if
else
     $p \leftarrow p_3$ 
end if
    
```

³essentia.upf.edu/models.html#discogs-effnet

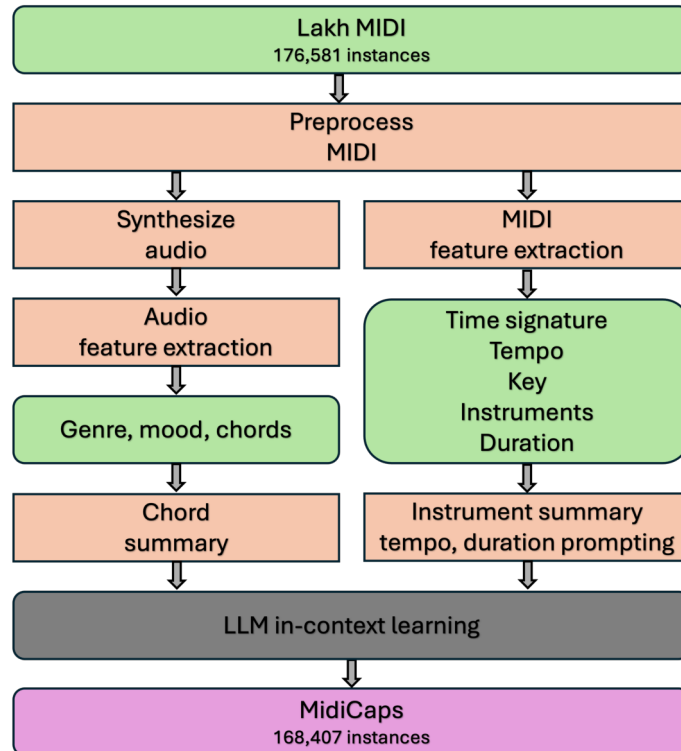


Figure 2: Detailed overview of our proposed captioning framework.

3.2 Caption generation

In this step, we take the extracted features and execute caption generation. To harness the expressive power and few-shot capability of a Large Language Model (LLM), we refer to a recent benchmarking study on LLMs [30], and ultimately selected Claude 3 Opus [31] due to its superior performance compared to other LLMs such as GPT4. Employing in-context learning (a task in which the LLM is given example data of paired input-output to serve as ‘context’, and is expected to continue producing outputs for new unpaired inputs in a similar manner), we begin by selecting 17⁴ diverse examples from the extracted features and request a human annotator to craft appropriate text captions for each of these based solely on the extracted features. This approach aims to prevent any auditory influence on human captioning, as Claude 3 (or any LLM, for that matter) will subsequently only process text inputs, not audio files. Once the 17 examples are prepared, we construct a text prompt instructing Claude 3 to analyze the human-prepared feature-caption pairs and generate suitable captions for new sets of features. To maintain clarity, we specify that the generated captions should be between three to seven sentences. Before generating captions for all 168K MIDI files, we conduct a sanity check on ten examples to evaluate Claude 3’s response to in-context learning, ensuring our prompt does not produce unrelated output or “hallucinate.” Please note, this check differs from the quality evaluation of the generated captions reported in the next section. In our study, a single round of sanity checks sufficed, obviating the need to modify prompts or alter the feature-caption pairs for in-context learning. Finally, us-

⁴Optimized based on limit on input tokens in Claude 3 text prompts.

ing the features extracted from each MIDI file, we generate corresponding captions, creating our proposed **MidiCaps** dataset, which we describe in detail in the next section.

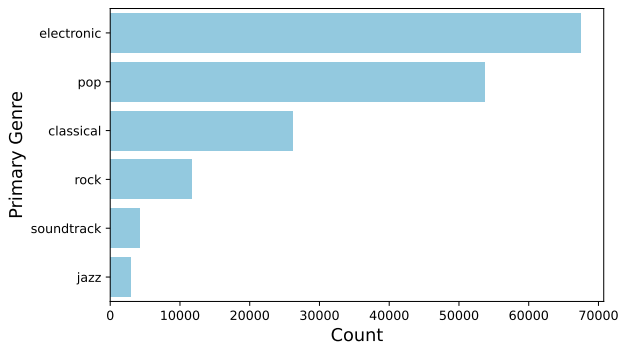
4. EVALUATION AND STATISTICS

In this section, we first introduce the **MidiCaps** dataset and subsequently detail subjective evaluation in form of listening study.

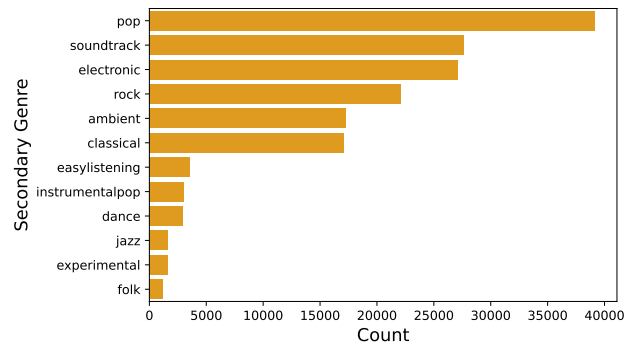
4.1 MidiCaps dataset

To generate our **MidiCaps** dataset, we start with MIDI files provided in the Lakh MIDI dataset [11], comprised of a collection of 176,581 unique MIDI files, designed to facilitate large-scale music information retrieval. Additionally, the dataset is distributed under a CC-BY 4.0 license, enabling us to expand the dataset without encountering copyright constraints. Subsequently, we process the raw MIDI files and extract musical features as described in Section 3.1, which we used in the captioning process Section 3.2 to create our final **MidiCaps** dataset consisting of 168,407 MIDI files with matching text caption. A couple of examples of captions generated are provided below. They encapsulate key information regarding the music contents while infusing a fluid human touch:

1. “A melodic and happy rock and pop song featuring a string ensemble, piano, clean electric guitar, slap bass, and drums. The song is in the key of F major with a 4/4 time signature and a tempo of 120 BPM. The chord progression alternates between Bb and F throughout the song, creating a motivational and energetic corporate vibe.”

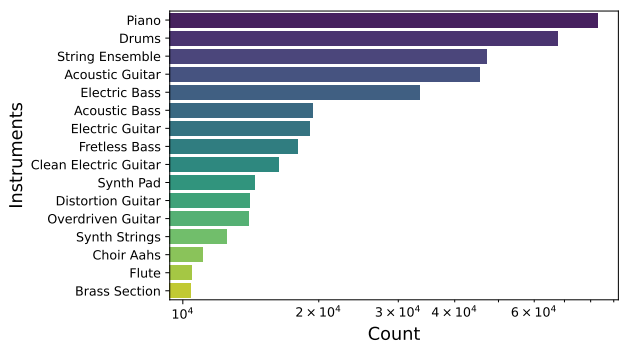


(a) Distribution of primary genre.

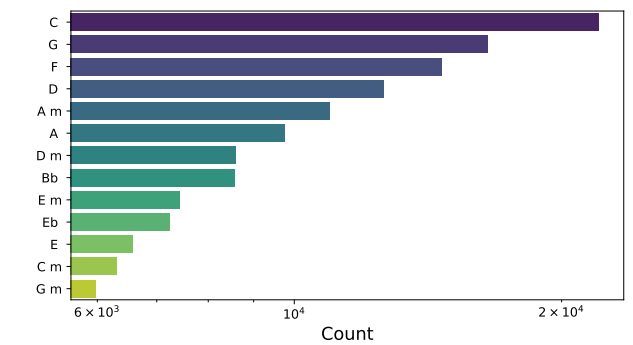


(b) Distribution of secondary genre.

Figure 3: Genre distributions.

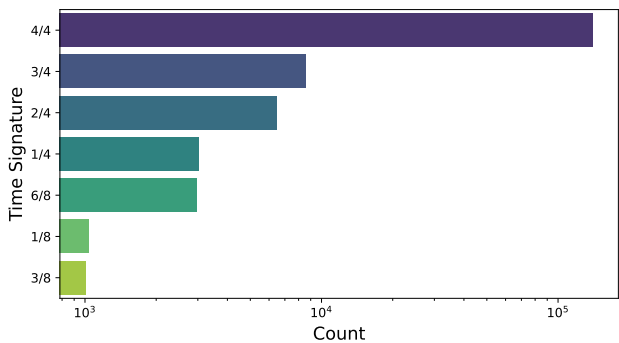


(a) Distribution of instruments present in the summary.

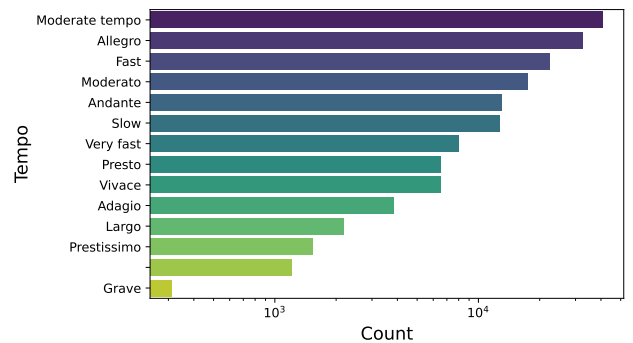


(b) Distribution of key.

Figure 4: Instrument and key distributions (in log scale).



(a) Distribution of time signature.



(b) Distribution of tempo.

Figure 5: Time signature and tempo distributions (in log scale).

2. “A melodic and happy pop song with a Christmas vibe, featuring piano, clean electric guitar, acoustic guitar, and overdriven guitar. The song is in the key of A major with a 4/4 time signature and a moderate tempo. The chord progression revolves around D, E6, D, and E, creating a motivational and loving atmosphere throughout the piece.”

Moreover, we provide a summary of some of the extracted features below to gain further insight into the diversity within the dataset. In Figure 3a, we illustrate the distribution of the primary (highest confidence score) and secondary (second highest confidence score) genres present in the dataset. In both cases, electronic and pop genres are

most prominent in the dataset. The secondary genre exhibits more variation, such as folk, instrumental pop, and easy listening, which have more occurrences as secondary genre but do not appear in the primary genre figure. This means that they can be used by the captioning system to further specify and narrow down the broad primary genres (e.g. classical) into more specific descriptions such as ‘ambient classical’ etc. Please note that only genres with more than 1,000 occurrences are displayed in the figures. Figure 4 summarizes the instruments and keys present in the dataset. Piano, drums, and various types of guitars are predominant in the instrument summary, corroborating the fact that a significant portion of the songs belongs to electronic, pop, and rock genres. Additionally, the keys

Audience: Annotated by:	General audience		Music experts	
	Human	AI	Human	AI
Question	Avg. rating (1-7)			
Overall matching	5.46	5.63	5.40	4.92
Human-like	5.21	5.32	5.09	4.98
Genre matching	5.80	5.63	5.54	4.73
Mood matching	5.50	5.62	5.43	4.82
Key matching	5.87	5.70	5.51	5.69
Chord matching	6.12	5.78	5.74	5.09
Tempo matching	5.71	5.86	5.37	5.77

Table 1: Results of the listening study. Each question is rated on a Likert scale from 1 (very bad) to 7 (very good). The table shows the average ratings per question for each group of participants.

of C, G, F, and D major have the highest occurrences in the dataset. Regarding time signature, 4/4 is significantly more common than any other (Figure 5a), while most songs follow a moderate tempo (Figure 5b).

4.2 Listening study

Since there is no ground truth or baseline model to compare our new dataset to, we conduct a listening study with the help of the PsyToolkit platform [32,33]. Listeners were asked to listen to 20 MIDI files, chosen at random, from which 15 are captioned by our framework and 5 are annotated by an expert human rater with absolute pitch. Then, listeners were asked to rate these captions in seven aspects, which are: 1) Overall matching of caption to audio, 2) How human-like the caption is, 3) Genre matching of caption with audio, 4) Mood matching, 5) Key matching, 6) Chord matching, and 7) Tempo matching. Those listeners who indicated that they do not have the ability to recognize chords/key were tagged as General audience. A total of 16 participants belong to this general audience, of which 25% has more than 1 year of musical training. Another 7 participants indicated that they can recognize chords and key or have absolute pitch. These were tagged as Music experts.

4.3 Results and discussion

Table 1 shows the results of the listening study. The average rating for overall matching of the text caption with the MIDI file for the general audience is even slightly higher (5.63) for the AI generated caption compared to the human-written caption (5.46). When it comes to the ratings by music experts, the overall matching rating is slightly lower, but still well above average (4.92). In term of how human-like the captions are, the general audience again provides high ratings, comparable to those given to the human-written captions (5.21). The music experts are slightly more critical and rate them at 4.98, which is still very close to their rating for human-written captions (5.09). A similar pattern can be seen for ratings of genre matching and mood matching. The ratings for tempo matching outperformed the human-written ones for both general audience and music experts.

In terms of key and chord matching, the general audience provide good ratings. For these questions the ratings from the music experts, however, are more reliable, as these participants have explicitly indicated that they are able to recognize chords and keys. Their rating for key matching (5.69) is on par with the rating for human-written captions (5.51), and confirm the high agreement that the musical key described in the caption matches the audio pieces. For chord matching, the music experts’ average rating of 5.09 falls below the rating for the human-written caption. Please note, however, that this particular question was not easy to answer. Extracting a single ‘main’ pattern (3-5 chords) from the entire list of extracted chords is challenging as there are many different cases, e.g., very short fragments of a few chords, and very long pieces with many chord patterns. Slight changes in chord patterns can also be intentional, e.g., a chord progression of C, G, D, C, G, D6 would likely be detected as a C, G, D, C, G pattern instead of a C, G, D variation. All this makes it hard to objectively judge a single-chord pattern in the text captions. Despite this, the chord matching rating of 5.09 provides support that our caption contains a matching chord summary. Overall, the results from the listening study support that our text captions provide a high-quality, human-like textual description that matches the MIDI files well.

The task of automatically labelling files of various length is difficult by nature as longer music pieces might require more text to be described precisely, while shorter pieces may need only a single sentence. This problem is further magnified when considering chord progressions and their summary as mentioned above. Additionally, extracting features from synthesized audio files is not optimal, as the choice of the sound font has an impact on the obtained results, which is likely to be most apparent in genre and mood features. Future research could focus on improving accuracy related to these features. In sum, we are confident that our **MidiCaps** dataset will facilitate the development of the first Text-to-MIDI generation models.

5. CONCLUSION

We present the first large-scale open MIDI captioned dataset, **MidiCaps**. This dataset also includes a comprehensive set of musical features such as chord patterns, genre, and mood. To facilitate the development of this dataset, we have developed a MIDI captioning framework. This approach includes music feature extraction and summarization from MIDI and the synthesized audio, as well as the use of the Claude-3 LLM to generate the final captions using in-context learning. To evaluate the final dataset, we have conducted two subjective listening studies, which confirm that the captions are natural and indeed contain a text description of the musical features contained in the accompanying MIDI file. The resulting new **MidiCaps** dataset contains 168,407 MIDI files with descriptive text captions and is available online⁵ under a Creative Commons licence.

⁵huggingface.co/datasets/amaai-lab/MidiCaps

6. ACKNOWLEDGMENTS

This project has received funding from the SUTD Kick-starter Initiative no. SKI 2021_04_06.

7. REFERENCES

- [1] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, “Llama: Open and efficient foundation language models,” *arXiv preprint arXiv:2302.13971*, 2023.
- [2] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, “Zero-shot text-to-image generation,” in *International conference on machine learning*. Pmlr, 2021, pp. 8821–8831.
- [3] J. Melechovsky, Z. Guo, D. Ghosal, N. Majumder, D. Herremans, and S. Poria, “Mustango: Toward controllable text-to-music generation,” *Proc. of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2024.
- [4] M. Schedl, E. Gómez, J. Urbano *et al.*, “Music information retrieval: Recent developments and applications,” *Foundations and Trends® in Information Retrieval*, vol. 8, no. 2-3, pp. 127–261, 2014.
- [5] D. Herremans, C.-H. Chuan, and E. Chew, “A functional taxonomy of music generation systems,” *ACM Computing Surveys (CSUR)*, vol. 50, no. 5, pp. 1–30, 2017.
- [6] A. Agostinelli, T. I. Denk, Z. Borsos, J. Engel, M. Verzetti, A. Caillon, Q. Huang, A. Jansen, A. Roberts, M. Tagliasacchi *et al.*, “Musiclm: Generating music from text,” *arXiv preprint arXiv:2301.11325*, 2023.
- [7] Q. Huang, D. S. Park, T. Wang, T. I. Denk, A. Ly, N. Chen, Z. Zhang, Z. Zhang, J. Yu, C. Frank *et al.*, “Noise2music: Text-conditioned music generation with diffusion models,” *arXiv preprint arXiv:2302.03917*, 2023.
- [8] H. Liu, Q. Tian, Y. Yuan, X. Liu, X. Mei, Q. Kong, Y. Wang, W. Wang, Y. Wang, and M. D. Plumbley, “Audioldm 2: Learning holistic audio generation with self-supervised pretraining,” *arXiv preprint arXiv:2308.05734*, 2023.
- [9] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez, “Simple and controllable music generation,” *arXiv preprint arXiv:2306.05284*, 2023.
- [10] S. Doh, K. Choi, J. Lee, and J. Nam, “Lp-musiccaps: Llm-based pseudo music captioning,” *arXiv preprint arXiv:2307.16372*, 2023.
- [11] C. Raffel, *Learning-based methods for comparing sequences, with applications to audio-to-midi alignment and matching*. Columbia University, 2016.
- [12] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler *et al.*, “Emergent abilities of large language models,” *arXiv preprint arXiv:2206.07682*, 2022.
- [13] C. Hawthorne, A. Stasyuk, A. Roberts, I. Simon, C.-Z. A. Huang, S. Dieleman, E. Elsen, J. Engel, and D. Eck, “Enabling factorized piano music modeling and generation with the maestro dataset,” *arXiv preprint arXiv:1810.12247*, 2018.
- [14] H.-W. Dong, W.-Y. Hsiao, L.-C. Yang, and Y.-H. Yang, “Musegan: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.
- [15] A. Ycart, E. Benetos *et al.*, “A-maps: Augmented maps dataset with rhythm and key annotations,” 2018.
- [16] F. Simonetta, F. Carnovalini, N. Orio, and A. Rodà, “Symbolic music similarity through a graph-based representation,” in *Proceedings of the Audio Mostly 2018 on Sound in Immersion and Emotion*, 2018, pp. 1–7.
- [17] S. Wu, D. Yu, X. Tan, and M. Sun, “Clamp: Contrastive language-music pre-training for cross-modal symbolic music information retrieval,” *Proc. of ISMIR*, 2023.
- [18] I. Manco, E. Benetos, E. Quinton, and G. Fazekas, “Muscaps: Generating captions for music audio,” in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021, pp. 1–8.
- [19] Q. Huang, A. Jansen, J. Lee, R. Ganti, J. Y. Li, and D. P. W. Ellis, “Mulan: A joint embedding of music audio and natural language,” in *Proceedings of the 23rd International Society for Music Information Retrieval Conference, ISMIR 2022, Bengaluru, India, December 4-8, 2022*, P. Rao, H. A. Murthy, A. Srinivasamurthy, R. M. Bittner, R. C. Repetto, M. Goto, X. Serra, and M. Miron, Eds., 2022, pp. 559–566. [Online]. Available: <https://archives.ismir.net/ismir2022/paper/000067.pdf>
- [20] Z. Kuang, S. Zong, J. Zhang, J. Chen, and H. Liu, “Music-to-text synaesthesia: Generating descriptive text from music recordings,” 2022.
- [21] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, “Training language models to follow instructions with human feedback,” *Advances in neural information processing systems*, vol. 35, pp. 27 730–27 744, 2022.
- [22] OpenAI, “Introducing ChatGPT,” 2023. [Online]. Available: <https://openai.com/blog/chatgpt>
- [23] M. contributors, “Mido: MIDI objects for python,” 2024. [Online]. Available: <https://github.com/mido/mido>

- [24] M. S. Cuthbert and C. Ariza, “music21: A toolkit for computer-aided musicology and symbolic music data,” 2010.
- [25] B. Zamecnik, “midi2audio.” [Online]. Available: <https://github.com/bzamecnik/midi2audio>
- [26] J. Newmarch and J. Newmarch, “Fluidsynth,” *Linux Sound Programming*, pp. 351–353, 2017.
- [27] FluidSynth Contributors, “FluidSynth: A real-time software synthesizer,” 2024. [Online]. Available: <https://github.com/FluidSynth/fluidsynth>
- [28] D. Bogdanov, N. Wack, E. Gómez Gutiérrez, S. Gulati, H. Boyer, O. Mayor, G. Roma Trepat, J. Salamon, J. R. Zapata González, X. Serra *et al.*, “Essentia: An audio analysis library for music information retrieval,” in *Britto A, Gouyon F, Dixon S, editors. 14th Conference of the International Society for Music Information Retrieval (ISMIR); 2013 Nov 4-8; Curitiba, Brazil.[place unknown]: ISMIR; 2013. p. 493-8. International Society for Music Information Retrieval (ISMIR), 2013.*
- [29] M. Mauch and S. Dixon, “Approximate note transcription for the improved identification of difficult chords,” in *Proceedings of the 11th International Society for Music Information Retrieval Conference, ISMIR 2010, Utrecht, Netherlands, August 9-13, 2010*, J. S. Downie and R. C. Veltkamp, Eds. International Society for Music Information Retrieval, 2010, pp. 135–140. [Online]. Available: <http://ismir2010.ismir.net/proceedings/ismir2010-25.pdf>
- [30] D. Kevian, U. Syed, X. Guo, A. Havens, G. Dullerud, P. Seiler, L. Qin, and B. Hu, “Capabilities of large language models in control engineering: A benchmark study on gpt-4, claude 3 opus, and gemini 1.0 ultra,” *arXiv preprint arXiv:2404.03647*, 2024.
- [31] Anthropic, “Claude 3 opus,” 2024. [Online]. Available: <https://www.anthropic.com/claude>
- [32] G. Stoet, “Psytoolkit: A software package for programming psychological experiments using linux,” *Behavior research methods*, vol. 42, pp. 1096–1104, 2010.
- [33] —, “Psytoolkit: A novel web-based method for running online questionnaires and reaction-time experiments,” *Teaching of Psychology*, vol. 44, no. 1, pp. 24–31, 2017.