A FOUNDATION MODEL FOR MUSIC INFORMATICS

Minz Won, Yun-Ning Hung, and Duc Le

SAMI, ByteDance, San Jose, CA, USA

ABSTRACT

This paper investigates foundation models tailored for music informatics, a domain currently challenged by the scarcity of labeled data and generalization issues. To this end, we conduct an in-depth comparative study among various foundation model variants, examining key determinants such as model architectures, tokenization methods, temporal resolution, data, and model scalability. This research aims to bridge the existing knowledge gap by elucidating how these individual factors contribute to the success of foundation models in music informatics. Employing a careful evaluation framework, we assess the performance of these models across diverse downstream tasks in music information retrieval, with a particular focus on token-level and sequence-level classification. Our results reveal that our model demonstrates robust performance, surpassing existing models in specific key metrics. These findings contribute to the understanding of self-supervised learning in music informatics and pave the way for developing more effective and versatile foundation models in the field. A pretrained version of our model is publicly available to foster reproducibility and future research.

***Index Terms*—** Foundation model, Music information retrieval, Self-supervised learning

1. INTRODUCTION

A foundation model refers to any pretrained machine learning model capable of being adapted to a broad spectrum of downstream tasks [1]. By taking advantage of its self-supervised nature, AI researchers have been able to scale up foundation models with enormous data, resulting in versatile representation that can generalize to diverse downstream tasks in multiple domains, such as natural language processing [2, 3], computer vision [4, 5], speech recognition [6, 7], and multimodal research [8].

The potential of foundation models is not only limited to the aforementioned domains. In the field of music information retrieval (MIR), researchers have consistently faced challenges in scaling up their models, mainly due to the lack of labeled data. The task of annotating music data is labor-intensive and often demands specialized domain knowledge, which hinders large-scale data collection efforts. To circumvent these limitations, researchers have employed semi-supervised learning [9, 10] or transfer learning [11, 12] schemes, which leverage knowledge from larger labeled datasets. Nevertheless, these approaches often encounter generalization issues, particularly when the downstream tasks involve information not represented in the supervised data [13]. Also, scalability remains constrained by the availability of labeled data. This has led to a growing interest in developing self-supervised foundation models specifically tailored for music informatics research.

Although research on foundation models in music informatics is still in its nascent stages, several pioneering works have made notable contributions. CLMR [14] and MULE [13] have employed a simple contrastive learning framework [15] to capture sequence-level music representation that summarizes the entire sequence, rather than preserving information at every time step. While their performance lags behind that of supervised methods, they have nonetheless demonstrated the potential for self-supervised models to generalize across multiple music tagging tasks. Another innovative study [16] has revealed that language models pretrained on tokenized representations, such as Jukebox [17], can serve as robust foundation models for various downstream MIR tasks. The authors discussed that the generative model could learn richer representations than conventional tagging models. More recently, MERT [18] has adapted HuBERT [6], a successful speech recognition foundation model, to formulate a framework explicitly tailored for music representation, proving its generalizability across an array of sequence-level classification tasks.

However, despite these advances, it remains unclear how individual factors, such as model architectures, tokenization methods, temporal resolution, data, and model scalability, contribute to the success of foundation models in music informatics. This knowledge gap underscores the need for a comprehensive investigation, which is the focus of our research. In this work, we compare a new self-supervised learning approach from speech recognition (i.e., BEST-RQ [7]) with MERT [18] through a meticulous evaluation of both token-level and sequence-level downstream classification tasks. We find that our model exhibits robust performance across a range of MIR tasks and outperforms existing models in specific contexts. A pretrained version of our model is available online [1] to facilitate reproducible research.

[1] https://github.com/minzwon/musicfm

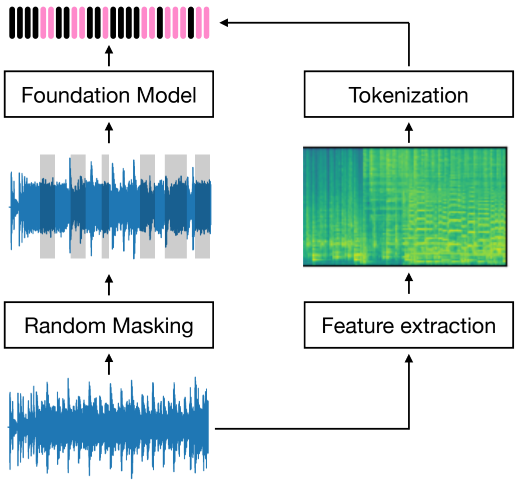

Fig. 1: Masked token modeling of audio representation.

2. MODELS

This section outlines the key concepts of foundation models that are employed in this study. Given that the objective of our research is to examine the various factors that contribute to the successful development of foundation models, we leverage existing models and training methodologies from previous works [7, 18]. It is important to note that none of the methods described in this section represent our original contributions; they serve as the basis for our investigations into optimizing foundation models for music informatics.

2.1. Masked token modeling

Alongside generative models such as GPT-3 [3], masked token modeling techniques, such as BERT [2], have shown robust performance as foundation models across a variety of sequential data types, particularly in speech audio [19, 20, 6, 7]. In the masked token modeling paradigm, the objective is to predict tokens that have been deliberately masked within a sequence. As illustrated in Figure 1, portions of the input audio are randomly masked with noise. The foundation model is then tasked with predicting these intentionally masked or omitted tokens (highlighted in pink). This approach enables the foundation model to learn contextualized semantics, which is useful for various downstream tasks. While various model architectures can serve as the backbone for a foundation model, transformer variants [19, 20, 6, 7] are most commonly used due to their exceptional sequence modeling capabilities. Specifically, the BERT-style encoder [6] has been incorporated into the MERT [18], and the Conformer [20] has been utilized in BEST-RQ [7], which are core backbones of our experiments. Masked token modeling can be formalized as a self-supervised classification task using the tokenization methods (right side of Figure 1) that will be introduced in the following subsection.

2.2. Tokenization

To cast music sequence modeling within the masked token modeling framework, it is essential to convert short audio segments into discrete tokens, a process known as tokenization. While various tokenization methods are available, our baseline, MERT [18], utilizes k-means clustering [6] and residual vector quantization (RVQ) [21]. The authors employ k-means clustering on log-mel spectra to capture timbral characteristics and chroma features for encoding harmonic attributes. The resulting representations are then tokenized according to their corresponding feature clusters.

However, most conventional tokenization methods, including k-means clustering and RVQ, necessitate a separate training phase for representation learning. This additional step can introduce complexities and create dependencies that may impact the foundation model's overall performance. To address these challenges, recent work (BEST-RQ [7]) proposed a tokenization method employing a random projection quantizer to bypass the need for a trainable representation learning phase. This approach has two random components: random projection and random codebook lookup, both of which obviate the need for training. Within this scheme, a d-dimensional input vector x is mapped to an h-dimensional latent space via random projection R. The closest index from a randomly initialized $n \times h$ codebook C is then selected as the feature-representing token. The tokenized representation τ can be formalized as:

$$\tau = \arg\min_i \left| \|c_i\|_2 - \|Rx\|_2 \right|, \tag{1}$$

where R is an $h \times d$ matrix for random projection, c_i represents the i-th vector in random codebook C, and $\|\cdot\|_2$ denotes the l_2-norm. The projection matrix R is initialized with Xavier initialization, while the codebook C uses standard normal distribution. Log-mel spectra serve as the only feature vectors and are normalized to have zero mean and unit variance. This normalization step is particularly crucial when employing a non-trainable random projection, as codebook utilization can be very low without it.

3. EXPERIMENTS

3.1. Datasets

We utilize two distinct datasets to train our foundation models. The first dataset consists of 160k hours of in-house music data, designed to align with the size of the data used to train MERT. The second dataset is the Free Music Archive (FMA) dataset [22], which comprises 8k hours of Creative Commons-licensed music audio. All audio files from both datasets have been preprocessed to a standard format of 24kHz mono audio.

3.2. Evaluation

Previous studies [14, 16, 13, 18] have largely focused on evaluating music foundation models through sequence-level classification tasks, such as genre classification, emotion recognition, key detection, and music tagging. However, many applications in music information retrieval necessitate predictions at individual time steps. Given this context, we propose that robust foundation models should demonstrate strong capabilities in token-level classification tasks, such as beat tracking and chord recognition. To test this hypothesis, our research evaluates foundation models across five distinct tasks: beat tracking, chord recognition, structure analysis (all token-level classification tasks), as well as key detection and music tagging (both sequence-level classification tasks).

Consistent with prior work [16], we employ a probing model on top of the foundation model for our evaluation. This probing model is structured as a shallow neural network with a single 512-dimensional hidden layer and an output layer. Since our goal is to scrutinize the pretrained representations, the foundation model remains frozen during this probing stage. In the case of sequence-level classification, we use an average pooling layer to aggregate the representations across time before initiating linear probing.

Beat/downbeat Tracking aims to predict the timestamps of each beat and the position of the first beat in each bar. In accordance with previous studies [23], our model generates frame-level probabilities of beats, downbeats, and non-beat events every 50 ms. To decode downbeat timestamps, we employ a dynamic Bayesian network (DBN) implemented in `madmom` [24] for post-processing. Harmonix Set [25] is used for training, while GTZAN [26] is used as a test set. We use F-measure implemented in `mir_eval` [27] for evaluation.

Chord Recognition is a challenging MIR task due to the intricate harmonic relationships within music compositions. In this work, we focus solely on major and minor chords. The model outputs frame-level probabilities of 25 classes, which include the 12 pitches of major and minor chords, along with one category denoted as "none," and does so at every 125 ms. We use HookTheory [28] for training, with 2000 songs selected for testing. The evaluation metric is major/minor weighted accuracy in `mir_eval` [27].

Structure Analysis aims to segment a music recording into distinct, non-overlapping sections and predict the functional label for each segment, such as 'verse' and 'chorus.' Following previous settings [29], the probing model has two classifiers designed to predict frame-level probabilities of seven functional classes and boundaries. Each frame has a resolution of 200 ms. We select 150 pieces from Harmonix Set [25] for testing, and the rest of Harmonix Set is used for training. We use the F-measure of hit rate at 0.5 seconds (*HR.5F*) to evaluate boundary and frame-wise accuracy to evaluate functional labels. Both metrics are computed using `mir_eval` [27] package.

Key Detection aims to predict the tonal scale and pitch relation across the entire songs. The model has to output frame-level probabilities of 25 classes (12 major and 12 minor keys plus one "none") per 2 seconds. We use HookTheory [28] for training. Giantsteps [30] is used as test set. The evaluation metric is refined accuracy (weighted accuracy) implemented by `mir_eval` [27], with error tolerance that gives partial credits to reasonable errors.

Music tagging subsumes various music classification tasks [31], such as genre, mood, instrument, and language classification. Since any musical attribute can be music tags, music tagging serves as a comprehensive downstream task to gauge a model's versatility. We employed the widely-used MagnaTagATune dataset [32], consisting of popular 50 tags. We maintained the same data splits as those employed in a previous study [33]. Evaluation metrics are the mean average precision (mAP) and the area under the receiver operating characteristic curve (ROC-AUC).

3.3. Foundation models

We implemented a self-supervised learning approach, as detailed in Section 2, which closely follows the methodology outlined in related work [7]. This approach takes advantage of random quantization, eliminating the requirement for separate representation learning. In line with the referenced paper, our codebook consists of 8192 16-dimensional vectors. We applied a 400ms window with a 60% probability mask, utilizing only the masked segment for cross-entropy loss optimization. FM8 in Table 1 follows the exact same setup of the previous work in the speech domain, BEST-RQ [7], except for the data.

In this study, we aim to address the fundamental question of how to build an advanced foundation model. To achieve this objective, we carefully scrutinize each component to assess their individual impacts. Our investigation encompasses an exploration of random quantization, an in-depth examination of various architectures, including a BERT-style encoder from HuBERT [6] and Conformer[20], while considering different model size configurations, as well as an investigation into varying temporal resolutions (i.e., the number of tokens per second), diverse input lengths for pretraining, and two distinct datasets. The summary of various model configurations can be found in Table 1.

We utilized the Adam optimizer [34] with a learning rate of 0.0001. Learning rate warm-up over 30,000 steps played a critical role in our training process, especially when using `float16`. To enhance training efficiency, we employed deepspeed [35], flash attention [36], and mixed precision techniques. All models were trained using eight A100-80GB GPUs for two weeks.

Table 1: Foundation model variants and their respective downstream metrics. FM1* corresponds to MERT [18], while FM8** mirrors the BEST-RQ [7] but with the distinction that it was trained using music data.

Foundation model							Beat		Chord	Structure		Key	Tagging	
Index	Encoder	Size	Hz	Input	Token	Data	Beat F1	Downbeat F1	Acc	Acc	HR.5	Acc	mAP	ROC
FM1*	BERT	330M	75Hz	5s	K-means	In-house	0.858	0.722	0.574	0.578	0.626	0.645	0.4499	0.9167
FM2	BERT	330M	75Hz	5s	Random	In-house	0.856	0.669	0.636	0.588	0.490	0.636	0.4185	0.9013
FM3	BERT	330M	75Hz	30s	Random	In-house	0.855	0.703	0.651	0.640	0.641	0.662	0.4226	0.9000
FM4	Conformer	330M	75Hz	5s	Random	In-house	0.863	0.771	0.643	0.619	0.534	0.660	0.4589	0.9161
FM5	Conformer	330M	75Hz	30s	Random	In-house	0.865	0.780	0.702	0.715	0.710	0.670	0.4821	0.9208
FM6	Conformer	330M	50Hz	30s	Random	In-house	0.864	0.802	0.690	0.710	0.698	0.671	0.4816	0.9204
FM7	Conformer	330M	25Hz	30s	Random	In-house	0.868	**0.804**	**0.714**	**0.726**	0.699	0.674	**0.4883**	**0.9235**
FM8**	Conformer	660M	25Hz	30s	Random	In-house	0.866	0.800	0.689	0.722	**0.716**	0.649	0.4790	0.9216
FM9	Conformer	330M	25Hz	30s	Random	FMA	0.868	0.767	0.675	0.664	0.631	0.674	0.4726	0.9167
FM7-finetune	Conformer	330M	25Hz	30s	Random	In-house	0.865	0.803	**0.807**	**0.742**	0.740	0.715	0.4809	0.9198
FM8-finetune	Conformer	660M	25Hz	30s	Random	In-house	0.874	**0.810**	0.800	0.739	**0.744**	0.700	0.4735	0.9192
FM9-finetune	Conformer	330M	25Hz	30s	Random	FMA	0.861	0.785	0.784	0.718	0.737	0.690	0.4695	0.9168
State-of-the-art [23, 37, 38, 29, 39, 40]							**0.887**	0.756	0.762	0.723	0.660	**0.746**	0.470	0.913

4. RESULTS

This section presents our experimental results and findings. Our carefully designed ablation study elucidates the critical factors of developing successful foundation models for music.

Random tokenization [7] generalizes well to music data. Despite the absence of an additional representation learning stage, it effectively acquires useful representations for various downstream tasks. Notably, even when random tokenization is exclusively applied to mel spectra, it surpasses FM1 (MERT) in the chord recognition task. This achievement is particularly noteworthy given that FM1 leverages an additional auxiliary task involving the constant-Q transform (CQT) reconstruction for capturing harmonic information. In the structure analysis task, an initial performance gap exists between FM1 and FM2, attributed to FM2's omission of auxiliary tasks, this gap is effectively bridged by employing longer input sequences (FM3).

Token-level classification offers a more comprehensive understanding of foundation models. When we solely compare various models using sequence-level classification, such as music tagging, it can be challenging to discern significant differences among them. However, token-level classification tasks, particularly those that demand a longer-term context, such as downbeat tracking and structure analysis, distinctly expose the limitations of models trained on shorter input sequences (FM2 and FM4).

Input length used during training is critical for capturing long-term contexts. Foundation models pretrained with 5s inputs (FM1, FM2, and FM4) excel in tasks related to timbre, such as music tagging, or tasks that only rely on local contexts, such as beat tracking. However, they exhibit lower performance than models trained with 30s inputs in tasks like downbeat tracking and structure analysis. We believe that advanced foundation models should be capable of modeling both long-term and short-term contexts, making longer inputs a recommended choice.

Temporal resolution has less impact in our experimental setup. A model with 25Hz temporal resolution (FM7) demonstrated slightly superior performance compared to its 50Hz (FM6) and 75Hz (FM5) counterparts, while demanding less computation due to shorter sequence lengths.

Model architecture makes a significant difference. Conformer (FM5) consistently outperformed BERT encoder (FM3) for across all downstream tasks. Interestingly, the influence of model size was relatively minimal (FM7 and FM8). However, it's worth noting that larger models might require longer training and more meticulous optimization to fully realize their performance potential.

Data is undeniably crucial, as in any data-driven approach. A model pretrained with 160k-hour music audio (FM7) showed better performance compared to a model trained on the 8k-hour FMA dataset (FM9). Two factors may contribute to this difference: first, the scalability, and second, the crowd-sourced nature of FMA, which encompasses a considerable amount of noisy data, impacting the model's generalizability.

Fine-tuning the foundation model further enhances downstream performance. However, we did observe a performance drop in the tagging task, primarily attributed to overfitting.

5. CONCLUSION

In this study, we conducted a thorough investigation into foundation model settings, evaluating their effects across five music information retrieval tasks, spanning both token-level and sequence-level classifications. Our experiments revealed six critical factors in the development of foundation models. As a result, our foundation model consistently surpasses its predecessor in all downstream tasks, with a notable performance gap, particularly evident in token-level classification tasks that necessitate long-term context. We anticipate that this enhanced foundation model will make valuable contributions, not only in classification tasks but also in the realms of multi-modal retrieval and generative models, as it incorporates a richer musical context.

6. REFERENCES

[1] Rishi Bommasani et al., “On the opportunities and risks of foundation models,” *ArXiv*, 2021.

[2] Jacob Devlin et al., “Bert: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. NAACL*, 2018.

[3] Tom Brown et al., “Language models are few-shot learners,” *Advances in neural information processing systems*, 2020.

[4] Alexey Dosovitskiy et al., “An image is worth 16x16 words: Transformers for image recognition at scale,” in *ICLR*, 2021.

[5] Anurag Arnab et al., “Vivit: A video vision transformer,” in *Proc. ICCV*, 2021.

[6] Wei-Ning Hsu et al., “Hubert: Self-supervised speech representation learning by masked prediction of hidden units,” *TASLP*, vol. 29, 2021.

[7] Chung-Cheng Chiu et al., “Self-supervised learning with random-projection quantizer for speech recognition,” in *Proc. ICML*, 2022.

[8] Alec Radford et al., “Learning transferable visual models from natural language supervision,” in *Proc. ICML*, 2021.

[9] Minz Won et al., “Semi-supervised music tagging transformer,” in *Proc. ISMIR*, 2021.

[10] Sangeun Kum et al., “Semi-supervised learning using teacher-student models for vocal melody extraction,” in *Proc. ISMIR*, 2020.

[11] Aäron Van Den Oord et al., “Transfer learning by supervised pre-training for audio-based music classification,” in *Proc. ISMIR*, 2014.

[12] Keunwoo Choi et al., “Transfer learning for music classification and regression tasks,” in *Proc. ISMIR*, 2017.

[13] Matthew C McCallum et al., “Supervised and unsupervised learning of audio representations for music understanding,” in *Proc. ISMIR*, 2022.

[14] Janne Spijkervet et al., “Contrastive learning of musical representations,” in *Proc. ISMIR*, 2021.

[15] Ting Chen et al., “A simple framework for contrastive learning of visual representations,” in *Proc. ICML*, 2020.

[16] Rodrigo Castellon et al., “Codified audio language modeling learns useful representations for music information retrieval,” in *Proc. ISMIR*, 2021.

[17] Prafulla Dhariwal et al., “Jukebox: A generative model for music,” *arXiv preprint arXiv:2005.00341*, 2020.

[18] Yizhi Li et al., “Mert: Acoustic music understanding model with large-scale self-supervised training,” *arXiv preprint arXiv:2306.00107*, 2023.

[19] Alexei Baevski et al., “wav2vec 2.0: A framework for self-supervised learning of speech representations,” *Advances in neural information processing systems*, 2020.

[20] Yu Zhang et al., “Pushing the limits of semi-supervised learning for automatic speech recognition,” *NeurIPS SAS Workshop*, 2020.

[21] Alexandre Défossez et al., “High fidelity neural audio compression,” *arXiv preprint arXiv:2210.13438*, 2022.

[22] Michaël Defferrard et al., “FMA: A dataset for music analysis,” in *Proc. ISMIR*, 2017.

[23] Yun-Ning Hung et al., “Modeling beats and downbeats with a time-frequency transformer,” in *Proc. ICASSP*, 2022.

[24] Sebastian Böck et al., “Joint beat and downbeat tracking with recurrent neural networks,” in *Proc. ISMIR*, 2016.

[25] Oriol Nieto et al., “The Harmonix Set: Beats, downbeats, and functional segment annotations of western popular music,” in *Proc. ISMIR*, 2019.

[26] Ugo Marchand et al., “Gtzan-rhythm: Extending the gtzan test-set with beat, downbeat and swing annotations,” in *ISMIR Late Breaking and Demo*, 2015.

[27] Colin Raffel et al., “Mir_eval: A transparent implementation of common mir metrics,” in *Proc. ISMIR*, 2014.

[28] Sangeun Kum et al., “Semi-supervised learning using teacher-student models for vocal melody extraction,” in *Proc. ISMIR*, 2020.

[29] Ju-Chiang Wang et al., “To catch a chorus, verse, intro, or anything else: Analyzing a song with structural functions,” in *Proc. ICASSP*, 2022.

[30] Peter Knees et al., “Two data sets for tempo estimation and key detection in electronic dance music annotated from user corrections,” in *Proc. ISMIR*, 2015.

[31] Minz Won et al., “Music classification: beyond supervised learning, towards real-world applications,” *arXiv preprint arXiv:2111.11636*, 2021.

[32] Edith Law et al., “Evaluation of algorithms using games: The case of music tagging.,” in *Proc. ISMIR*, 2009.

[33] Minz Won et al., “Evaluation of cnn-based automatic music tagging models,” in *Proc. SMC*, 2020.

[34] Diederik P Kingma et al., “Adam: A method for stochastic optimization,” in *ICLR*, 2015.

[35] Jeff Rasley et al., “Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters,” in *Proc. ACM SIGKDD*, 2020.

[36] Tri Dao et al., “Flashattention: Fast and memory-efficient exact attention with io-awareness,” *Advances in Neural Information Processing Systems*, 2022.

[37] Jonggwon Park et al., “A bi-directional transformer for musical chord recognition,” in *Proc. ISMIR*, 2019, pp. 620–627.

[38] Taejun Kim et al., “All-in-one metrical and functional structure analysis with neighborhood attentions on demixed audio,” in *Proc. WASPAA*, 2023.

[39] Filip Korzeniowski et al., “Genre-agnostic key classification with convolutional neural networks,” in *Proc. ISMIR*, 2018.

[40] Pablo Alonso-Jiménez et al., “Pre-training strategies using contrastive learning and playlist information for music classification and similarity,” in *Proc. ICASSP*, 2023.