## Unlocking Potential in Pre-Trained Music Language Models for Versatile Multi-Track Music Arrangement

Longshen Ou<sup>1</sup>, Jingwei Zhao<sup>1,2,3</sup>, Ziyu Wang<sup>4,5</sup>, Gus Xia<sup>4,5</sup>, Ye Wang<sup>1,2,3</sup>,

<sup>1</sup>Sound and Music Computing Lab, School of Computing, NUS

<sup>2</sup>Institute of Data Science, NUS

<sup>3</sup>Integrative Sciences and Engineering Programme, NUS Graduate School

<sup>4</sup>Music X Lab, MBZUAI

<sup>5</sup>NYU Shanghai

oulongshen@u.nus.edu, jzhao@u.nus.edu, zz2417@nyu.edu, gxia@nyu.edu, wangye@comp.nus.edu.sg

#### **Abstract**

Large language models have shown significant capabilities across various domains, including symbolic music generation. However, leveraging these pre-trained models for controllable music arrangement tasks, each requiring different forms of musical information as control, remains a novel challenge. In this paper, we propose a unified sequence-tosequence framework that enables the fine-tuning of a symbolic music language model for multiple multi-track arrangement tasks, including band arrangement, piano reduction, drum arrangement, and voice separation. Our experiments demonstrate that the proposed approach consistently achieves higher musical quality compared to task-specific baselines across all four tasks. Furthermore, through additional experiments on probing analysis, we show the pre-training phase equips the model with essential knowledge to understand musical conditions, which is hard to acquired solely through task-specific fine-tuning.

#### Introduction

Symbolic music arrangement is a crucial research area in generative modeling, focusing on how various music elements can be controlled by other musical information. These tasks significantly enhance the controllability and interpretability of automatic music generation. Current models in this field include those for lead melody and accompaniment arrangement (Zhao and Xia 2021; Wang, Min, and Xia 2024), chord generation (Simon, Morris, and Basu 2008; Yi et al. 2022), instrumentation (Zhao, Xia, and Wang 2023a), style transfer (Yang et al. 2019; Wang et al. 2020), track infilling (Malandro 2023), inpainting (Wei et al. 2022; Min et al. 2023), and more. These models are typically task-specific, incorporating music-tailored inductive biases in their architecture and data representation to achieve better controllability and interpretability.

As large symbolic music language models have shown significant improvements in generation quality (Qu et al. 2024), there is potential for these models to further enhance the current standard of automatic music arrangement. However, due to the task-specific design of arrangement mod-

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

<sup>1</sup>The code and demos are in the supplementary materials. We will open source them once accepted.

els, integrating large-scale pre-training can be less straightforward on those tasks. Further, the controllability of current music language models is primarily driven by text or a set of predefined attributes, which limits control to broad, coarse-grained features such as style, pitch range, and rhythmic density (Copet et al. 2024; Lu et al. 2023). Fine-grained control over musical content is not inherently supported by these models. The effectiveness of using fine-graind control in symbolic music language models remains to be explored.

This paper addresses this gap by focusing on multi-track symbolic music arrangement to explore how a weakly controllable pre-trained model can be adapted into a more controllable sequence-to-sequence model suitable for various arrangement tasks, which takes in complex multi-track music sequences as conditions, understand them, and generate desired outputs accordingly. We also show that generative pre-training provides a robust foundation for understanding conditions and generating music, as evidenced by both ablation studies and knowledge probing analyses. In summary, the contributions of this paper are as follows:

- We propose a universal fine-tuning objective for multitrack symbolic music arrangement. The design incoperates both global music attributes and fine-grained music content control, and is compatible with autoregressive language models. The objective is also extensible to other conditional music generation tasks.
- We achieve high-quality and versatile multi-track music arrangement with the pre-train and fine-tune paradigm. Experiments show our models consistently outperform state-of-the-art task-specific methods in four arrangement tasks: band arrangement, piano reduction, drum arrangement, and voice separation.
- We provide empirical analysis of necesity of pretraining in music arrangement tasks. Experiments demonstrate that models with pre-training outperform those without it. Further with knowledge probing, we show that pre-training help to acquire essential musical knowledge that are useful in the arrangement tasks, and such knowledge cannot be effectively learned through fine-tuning alone.

#### **Related Work**

Transformers in Symbolic Music Generation In recent years, the Transformer network has become increasingly prominent in various music generation tasks, particularly in conditional music generation, which involves generating music based on a given musical context (Ji, Yang, and Luo 2023). Transformers have been applied to tasks such as chord-to-melody generation (Madaghiele, Lisena, and Troncy 2021), accompaniment arrangement from lead sheets (Ren et al. 2020), and music inpainting (Malandro 2023). However, most works focus on controlled generation using simple high-level attributes like style (Choi et al. 2020), structure (Zhang et al. 2022), and sentiment (Makris, Agres, and Herremans 2021), with little exploration of using music itself as content conditions. Additionally, the efficacy of the pre-train and fine-tune scheme on conditional music generation remains underexplored.

Automatic Arrangement in Multi-Track Symbolic Music Research in automatic arrangement aims to adapt compositions to different instrumental settings while preserving the original intent. Earlier approaches typically used supervised methods to train sequence-to-sequence models with parallel, fixed-direction data, such as piano-to-orchestra (Crestel and Esling 2016) or band-to-piano (Terao et al. 2022). However, supervised training has significant drawbacks: crafting such datasets is challenging, and it restricts the model's flexibility by confining the arrangement direction to a fixed one. Alternative approaches include classification-based methods (Dong et al. 2021), where the model predicts the instrument label for each note from a flattened piano roll. However, this approach lacks creativity and limits the arrangement to a fixed set of instruments. Recent self-supervised methods, such as O&A (Zhao, Xia, and Wang 2023b,a), offer new possibilities. However, (Zhao, Xia, and Wang 2023b) assumes prior knowledge of the output distribution, which may not always be feasible, while (Zhao, Xia, and Wang 2023a) separates style and content modeling, potentially compromising the faithfulness of the arrangement. Additionally, Composer's Assistant (Malandro 2023) can handle arrangement tasks requiring creativity through infilling but may struggles to maintain consistency across different segments.

# **Prompt-based Fine-Tuning for Conditional Generation** In NLP, prompt-based fine-tuning has become a widely

adopted technique for controlled text generation (Liu et al. 2023). This approach uses specific control tokens to guide the output of a pre-trained language model, such as style tokens (Sennrich, Haddow, and Birch 2015), length specifications (Kikuchi et al. 2016), and pronunciation requirements (Ou et al. 2023). Among various pre-training objectives, the standard left-to-right language modeling is particularly well-suited for generation tasks (Radford et al. 2019; Brown et al. 2020). This setup allows the model to effectively utilize control tokens during fine-tuning, producing outputs that adhere to the specified conditions while maintaining fluency and relevance. The success of prompt-based fine-tuning in NLP offers valuable insights for designing our arrangement model, where similar pre-training methods and control mechanisms can be applied to guide the generation

process, ensuring that the output meets specific musical requirements while preserving coherence in the arrangement.

Knowledge Probing from Hidden Representations Knowledge probing techniques are used to explore what a model has learned within its hidden representations (Rogers, Kovaleva, and Rumshisky 2020). A widely adopted method is linear probing, where the pre-trained model is frozen and a simple linear classifier is trained on top of the hidden representations to predict specific properties, thereby revealing the extent to which particular knowledge is encoded in the model (Alain and Bengio 2016). This technique is particularly useful in our context for evaluating what the model has internalized during pre-training.

#### Method

We adopt a pre-train and fine-tune paradigm on the arrangement task. In the first stage, a Transformer decoder undergoes standard language model training on a large unlabeled corpus. Then, we fine-tune the model in a sequence-to-sequence manner with the proposed objectives on multiple arrangement tasks, including band arrangement, piano reduction, and drum arrangement, which are typical arrangement scenarios in music studies, and an additional voice separation task to test the model's instrument modeling. Despite the diversity of scenarios in music arrangement, as we will show later, our methodology remains a consistent better performance, over each task-specific SOTA models.

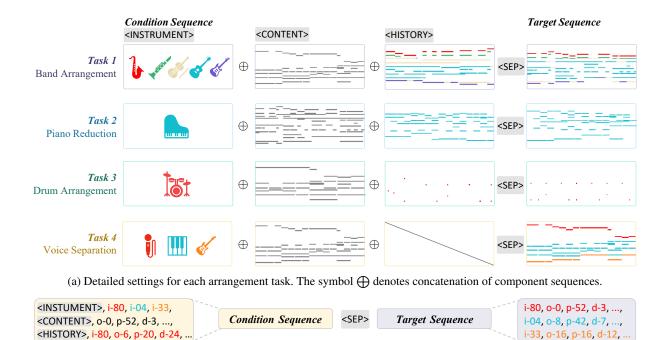
#### **REMI-z Representation**

We propose REMI-z, a variant of REMI+ (von Rütte et al. 2023) tokenization, and adopted it in both pre-training and fine-tuning. We adopted a simpler version of vocabulary where only five types of token were used: **instrument** (i-X), note's within-bar **position** (o-X), **pitch** (p-X), **duration** (d-X), with an additional **bar-line** token (b-1) used as a separator. Velocity tokens were not used for simplicity. For position and duration, we adopted same time quantization granularity as REMI+ (48th-note resolution).

In the tokenization process, an entire piece of music is represented by concatenation of *bar sequences* with b-1 as separator. Each bar sequence is a concatenation of *track sequences*, where the instrument tokens serve as the separator, and the order of track sequences is determined by their average pitch (from high to low). Each track sequence contain the information of all the notes played by a certain instrument within a single bar, and each note is represented by a triplet of (position, pitch, duration). Notes within a track sequence are sorted first by position (from left to right) and then by pitch (from high to low), and finally, flattened into a 1-d sequence.

## **Unconditional Pre-training**

Our pre-training process utilizes next-token prediction as the sole objective, employing REMI-z tokenization. For our experiments, we use a 12-layer GPT-2 model with 80M parameters, a relatively modest size that allows for full parameter fine-tuning. The pre-training dataset comprises approximately 4.3B tokens. While neither the model nor the dataset



(b) An example of the tokenized sequence for a band arrangement task. Special tokens <SEP>, <INSTRUMENT>, <CONTENT>, and <HISTORY> are used to structure the sequence and differentiate between components.

Figure 1: We address four distinct music arrangement task: 1) band arrangement, 2) piano reduction, 3) drum arrangement, and 4) voice separation. For each task, a music segment is reconceptualized using a specified set of instruments (highlighted with distinct colors), with the option to incorporate historical arrangement context to ensure long-term coherence. The given instruments, music segment, and history are tokenized to form a condition sequence. The target arrangement forms the target sequence. We train a sequence-to-sequence model based on next-token prediction, with cross-entropy loss computed from the target sequence.

is exceptionally large, this configuration has proven sufficient for achieving robust performance in music arrangement tasks. We hypothesize that further scaling up the model and dataset in pre-training would lead to even greater performance improvements.

#### **Arrangement Fine-Tuning**

Music arrangement involves interpreting musical ideas under a new set of instrumental constraints. Although at inference time, the instrumental constraints are usually differ from the original composition, the ability of interpreting given content with certain group of instruments can be modeled with a sequence reconstruction objective. Assuming  $y^{(t)}$  is the t-th segment in a certain piece of music, our proposed overall segment-level fine-tuning objective is:

$$\mathcal{L}(\theta) = -\log p_{\theta}(y^{(t)} | I(y^{(t)}), C(\text{aug}(y^{(t)})), y^{(t-1)}),$$

where  $\theta$  represents model's parameters, and  $I(\cdot)$  and  $C(\cdot)$  are two operators to extract *instrument* and *content* information from a given music segment respectively, while  $y^{(t-1)}$  offers target-side *history*. The training is implemented by standard LM objective on concatenated <code>[condition]<sep>[target]</code> sequence, with cross entropy loss computed only on the target subsequence. The entire sequence representation is illustrated in Figure 1b. At

training time, instrument and content conditions comes from a same music segment, while at inference time, instrument constraints can be flexibly set by user as desired, and content comes from the music to be arranged. We will elaborate on the three conditions, instrument, content, and history, as below.

Instrument conditions, resulted from  $I(\cdot)$ , specify the desired instruments to be used in the arrangement. In training, it is constructed using all used instrument tokens in the target-side sequence. Additionally, the tokens location inside the instrument sequence are used to embed another important information—the voice relationship between instruments. The more frontal part the instrument token locates, the higher pitch range it has for the instrument, and vice versa. At inference time, user has freedom to control what instrument to be used, together with their voice relationship, in segment-level.

We adopt *flattened piano rolls* as the content condition, inspired by (Zhao, Xia, and Wang 2023b). We also tokenized them into a sequence format, the steps are as below: (1) Removing all instrument tokens from the REMI-z, (2) Sort all notes strictly by their positions, (3) Sort notes within a same position by pitch (from high to low), and (4) Merge duplicated notes on same position and with same pitch. Further, a task-specific content augmentation  $\operatorname{aug}(\cdot)$  is applied in train-

ing to boost creativity, which will be elaborated when introducing each tasks.

Arranging a complete song in one pass is unfeasible with our model due to context length limitations of the model, hence we segment the original song and arranging each segment sequentially. To get rid of potential inconsistency issue between segments, an additional condition is adopted as input—target-side history of previous segment, inspired by (Tiedemann and Scherrer 2017). In training, history is provided by teacher-forcing (ground truth REMI-z of previous segment), while autoregressively at inference time, by using output of previous segment. In this way, consistent arrangement can be generated for entire pieces of music.

#### **Tasks**

We test the effectiveness of our method on the tasks below, which are all typical arrangement tasks in music composition studies. Below we introduce these tasks, their required model's capabilities each task assesses, and task-specific implementations.

**Band Arrangement** This task aims to arranging an existing piece of music for a new set of instruments. The model must understand the properties of each instrument and their common playing styles to allocate or create new notes appropriately. To foster creativity, we employ a strategy to deleting parts of content sequence that are from a groups of randomly selected tracks, to encourage the model using new notes that can be potentially compatible with the musical context but not provided in the content input at inference time. Such content deletion is defined to be aug in this task. During this process, special attention is given to preserving melody by ensuring the track with highest average pitch remains untouched during track deletions. Further, to enhance the naturalness when infer with new instrument sets, the duration tokens were deleted from the content sequence, to enable the model generate duration that is most suitable for the playing styles of each desired instruments, but not simply copy from input. The length of segment is set to 1 bar.

**Piano Reduction** Arranging an existing piece of music to a solo piano accompaniment. This task requires simplifying the original composition while identify and retaining its essential framework and important texture. When training with multi-track data, we use the original piano track as y, while the content sequence comes from the entire multi-track music, resulting the aug's definition as "adding notes that does not belong to y to the condition". To ensure the piano part is actually the dominant track in training data, we filtered the training data so that only the pitch range is larger than 0.4 times of the entire pitch range of the segment, which results in about 50% of the entire dataset. The segment length is also set to 1 bar.

**Drum Arrangement** Creating a drum track for songs that lack one. In training, the input sequence still contains instrument condition (only a drum token this time), content condition, and target-side history. Since REMI uses a unique set of pitch tokens for drums, providing the original drum content would lead to direct copying. Hence, aug is defined

as removing all original drum notes from the content condition. A longer segment length is adopted (4 bars), as drum patterns are more "flattened" along the time axis and require more context to infer a complete pattern.

Voice Separation This task aims to faithfully retrieve each instrumental track from the flattened piano roll. The model will be given the original instrument setting (unfixed number/types) and the original content sequence, and are asked to generate the content of each track as accurate as possible. This task reflect the understanding of the role of instruments in different ensemble settings, which serve as a foundation for other arrangement tasks. We cancel the aug operation to let it be a deterministic problem, where objective evaluation can be adopted. Further, the history was also canceled from the condition to remove any external hint to the generation.

#### **Experiments**

In this section, we evaluate the performance across four arrangement tasks. We begin by detailing the implementation, followed by a discussion of the metrics, baseline models, and results for each task.

#### **Implementation Details**

Our model, an 80M-parameter decoder-only Transformer, was trained solely on unconditional next-token prediction. It features a hidden representation dimension of 768, with 12 layers and a context length of 2048 tokens—roughly corresponding to eight bars in our tokenization scheme. During fine-tuning, the sample length is determined by the number of tokens within each segment. We employed the AdamW optimizer (Loshchilov, Hutter et al. 2017) with a cosine learning rate scheduler for pre-training and a linear learning rate scheduler with warm-up for fine-tuning. The learning rates were set to 5e-4 for pre-training and 1e-4 for finetuning. Pre-training was conducted using four RTX A5000 GPUs (24GB each), while fine-tuning was performed on a single A40 GPU (48GB). The batch size was 12 for pretraining and varied during fine-tuning, adjusted to the maximum value allowed by the available VRAM. Pre-training was completed in a single epoch, and fine-tuning was conducted over three epochs.

We used two datasets for pre-training and fine-tuning our model, respectively. Pre-training was conducted using the Los-Angeles-Project dataset (Lev 2024), a large-scale MIDI dataset comprising approximately 405K unique MIDI files, totaling around 4.3B tokens after applying REMI-z tokenization. To monitor training progress, 2% of the songs were randomly selected as a validation set. For fine-tuning, we utilized a smaller yet meticulously curated dataset, Slakh2100 (Manilow et al. 2019), which contains 1,289 MIDI files in the training set, 270 in the validation set, and 151 in the test set after deduplication. This dataset features a balanced distribution of 34 pitched instrument classes and an additional drum track, with each piece including at least five instrumental tracks along.

#### **Subjective Evaluation on Arrangement**

For the band, piano, and drum arrangement tasks, we opted for human evaluation since there is no definitive ground truth when arranging music with new instrument settings, and similarity metrics alone cannot accurately reflect quality, especially when creativity is involved. We generated full-piece arrangements using all models and methods, then asked human evaluators to compare them phrase-by-phrase, on a 5-point scale from 1 (very low) to 5 (very high). Additionally, for the band arrangement task, we tested the models' abilities across different scenarios by using multiple groups of instrument combinations, including string trio, rock band, and jazz band. Further details about the questionnaire and evaluation process are provided in the Appendix.

**Subjective Metrics** There are three metrics that are used across band, piano, and drum arrangements. They are:

- **Coherence**, which evaluates the natural flow of the arrangement and the consistency of each instrument's performance and style throughout the piece.
- Creativity, which assesses the degree of innovation in the arrangement under the constraints of content and style of the music; and
- **Musicality**, which evaluate the overall musicality of the arrangement.

In addition to the metrics above, there are several task-specific metrics. For band arrangement, **Faithfulness** is used to assess how closely the outputs resemble the original piece in terms of melody and overall feel. **Instrumentation** evaluates the appropriateness of each instrument's role within the band and whether they harmonize effectively. The piano reduction task also utilizes Faithfulness, along with a new metric called **Playability**, which evaluates how feasible it is for a human pianist to perform the accompaniment. For drum arrangement, two additional metrics are used: **Compatibility**, which assesses how well the drum sounds blending with other instruments when played together, and **Phrase Transition**, which evaluates how smoothly the drum arrangement handles transitions between phrases.

Baseline Models For each arrangement task, we use a previous state-of-the-art (SOTA) model as the primary baseline. Specifically, we compare against AccoMontage-3 (Zhao, Xia, and Wang 2023a) for band arrangement and piano reduction tasks, and Composer's Assistant (Malandro 2023) for the drum arrangement task. Additionally, all experiments include a comparison with an ablation variant of our model, where generative pre-training is omitted (No PT). We also incorporate rule-based algorithms as additional baselines. For the band arrangement task, the rule-based algorithm (Rule-B) distributes notes within each bar evenly by pitch, assigning them to different instruments based on their voice relationships. For piano reduction, we use two rule-based baselines: Rule-F (flattened), where a piano is required to play all notes in the flattened piano roll of the original composition, and Rule-O (original), which directly uses the original piano track from within a band as a baseline. Finally, for drum arrangement, the original drum track of a

Model	Fa.	Co.	In.	Cr.	Mu.
Rule-based	3.46	3.05	2.89	3.00	3.07
AccoMontage-3	2.65	2.70	2.72	3.00	2.72
Ours	3.77	3.47	3.49	3.40	3.47
No PT	3.19	2.82	2.86	2.93	2.75

Table 1: Band arrangement results. Fa., Co., In., Cr., and Mu. represent Faithfulness, Coherence, Instrumentation, Creativity, and Musicality, respectively.

Model	Fa.	Co.	Pl.	Cr.	Mu.
Rule-F	3.41	3.39	2.83	3.13	3.26
Rule-O	3.26	3.52	3.78	2.70	3.02
AccoMontage-3	3.28	3.59	3.72	3.26	3.26
Ours	3.65	3.87	3.96	3.83	3.78
No PT	2.52	2.74	3.48	3.04	2.72

Table 2: Piano reduction results. The Pl. represents Playability score.

piece of music (ground truth) is also included anonymously in the human evaluation.

Results on Band Arrangement Rule-based models demonstrated moderate performance across all metrics with scores consistently around the mid-3 range, highlighting a balanced but not outstanding approach to music arrangement. AccoMontage-3, serving as a primary baseline, generally scored lower than the Rule-based model, particularly in Faithfulness (2.65) and Musicality (2.72), suggesting limitations in maintaining the original essence of pieces and less appealing in musical expression. On the other hand, our model exhibited superior performance across all metrics, with notable improvements especially in Faithfulness (3.77) and Coherence (3.47). These results underscore our model's capability to not only adhere closely to the original musical content, but also ensure a smooth and logical progression of the arrangement, with appropriate creativity, and higher musicality. No Pretraining (No PT) configuration of our model showed reduced effectiveness, particularly in maintaining Coherence (2.82) and Musicality (2.75). This suggests the significant role of pretraining in enhancing the model's understanding and execution of complex musical arrangements.

In summary, our proposed model outperforms established baselines, validating the efficacy of our approach, particularly when pretraining is employed. The drop in performance in the No PT configuration highlights the critical importance of pretraining in achieving high-quality music arrangements.

Co.	Comp.	Tr.	Cr.	Mu.
4.09	3.98	3.55	3.53	3.98
3.13	2.87	2.51	2.43	2.64
3.81	3.87	3.77	3.40	3.60
2.66	2.74	2.64	2.64	2.57
	<b>4.09</b> 3.13 3.81	<b>4.09 3.98</b> 3.13 2.87 3.81 3.87	4.09     3.98     3.55       3.13     2.87     2.51       3.81     3.87     3.77	4.09     3.98     3.55     3.53       3.13     2.87     2.51     2.43       3.81     3.87     3.77     3.40

Table 3: Drum arrangement results. Comp. and Tr. represent Compatibility and Phrase Transition score respectively.

	Inst-related		Se	Segment-level		Track-level		Note-level
Model	I-IOU	V-WER	P-IOU	O-IOU	M-R	P-IOU	O-IOU	D-D
Q&A w/o func	97.50	35.01	60.40	86.96	18.84	43.53	70.69	0.29
Ours w/o voice	96.46	36.98	96.88	95.83	93.77	62.86	70.02	0.37
Ours	97.94	13.61	96.87	95.02	93.52	70.43	73.81	0.32
No PT REMI-z	92.05	39.75	97.02	92.99	94.19	57.93	65.24	0.40
No PT REMI+	53.16	51.73	60.53	54.98	30.12	43.37	33.34	0.45

Table 4: Voice separation results. All values in the table are expressed as percentages without the percent sign.

**Results on Piano Reduction** As in Table 2, our model excelled by achieving an optimal balance between faithfulness and playability, ensuring that the arrangements were technically feasible and accurately reflected the original compositions' emotional depth. This demonstrates our model's robust understanding of piano arrangement, effectively meeting the objectives of piano reduction. Additionally, our approach skillfully balanced coherence with creativity, resulting in musically rich arrangements that maintain a consistent playing style while incorporating unexpected elements, enhancing overall musicality. This distinct capability sets our model apart from others, which often less competetive on creativity and playability. Notably, models without pretraining displayed significant deficiencies in maintaining faithfulness and coherence, leading to lower overall musicality scores.

**Results on Drum Arrangement** As shown in Table 3, our model demonstrated significant advancements, closely approaching the performance of the ground truth across multiple metrics. It notably excelled in compatibility (3.87) and creativity (3.40), and even achieved a higher score than the ground truth in handling phrase transitions (3.77). This proficiency in managing phrase transitions indicates a deep understanding of the structural information from the input musical content. In contrast, the Composer's Assistant (T5) falls short in matching the structural coherence and creative drum fills produced by our model. The No PT model displayed limited abilities in all aspects, further underscoring the importance of pretraining in achieving high-quality drum arrangements. Notably, the performance gap between our model and the non-pre-trained model is larger, compared to other arrangement tasks, emphasizing the critical role of generative pretraining in handling longer sequences in downstream sequence-to-sequence tasks.

#### **Objective Evaluation on Voice Separation**

For the voice separation task, where the objective is to reconstruct the output as closely as possible to the reference REMI-z sequence, objective metrics alone are sufficient. These metrics will be introduced in this section, along with the baselines and evaluation results for the voice separation task.

**Objective Metrics** We utilize several specific metrics to gauge various aspects of performance, which are mainly based on word error rate (WER) and intersection over union (IoU): **Instrument IoU (I-IoU)** assesses the accuracy of instrument control. **Voice WER** (V-WER) measures the similarity in voice features between the generated output and

the reference. **Pitch IoU** (P-IoU) and **Position IoU** (O-IoU) are calculated to evaluate content similarity. These metrics are applied at both the segment level, to assess global content similarity, and the track level, where calculations focus on the overlap of instruments between the reference and the output. This latter measure is crucial as it directly reflects the model's understanding of the roles of different instruments. **Melody Recall** (M-R) is used to evaluate the fidelity of the melody reproduction. Lastly, the track-wise **absolute average duration difference** (D-D) quantifies the accuracy of duration prediction in the generated output, with the unit of beat. We will explain the detailed computation in the Appendix.

Baseline Models In this task, we use the Q&A model from (Zhao, Xia, and Wang 2023b) as the baseline. The original Q&A has access to the output's track-wise distribution on time and pitch axis (function) as prompt, which is a strong hint for the generation. For a fair comparison, they are removed during implementation (Q&A w/o func). Further, we include an ablation study where the voice information is removed from instrument prompt by sorting all instrument tokens by MIDI program ID (Ours w/o voice). We also compare the effectiveness of different tokenization schemes on the fine-tune-only experiments (No PT REMI-z and No PT REMI+).

**Evaluation Results** Table 4 shows the results of voice separation. Compared to the baseline, our model without voice control achieves a comparable Instrument IoU, demonstrating its effectiveness in meeting the instrument condition requirements. On segment-level metrics, our model consistently outperforms the baseline, especially showing the advantage of preserving melody notes during arrangement (M-R 93.77 vs. 18.84), which is crucial for the output fidelity. On track-level metrics, the baseline have comparable performance on predicting groove (i.e., position of notes). However, ours is much better at predicting the pitch for each track, showing better modeling of context-aware instrument styles. Additionally, despite not using any duration information from input, our model predicts duration with a comparable level of accuracy (-0.08), showing its good understanding of different playing styles of different instruments.

The results also illustrates the effectiveness of voice control. When adding the voice information in the instrument condition (Ours), the V-WER significantly drop, indicating that the instruments in the output has closer voice relationship compared to the target composition, meaning that the generation process can be effective guided by the voice information. It also makes some other metrics better, includ-

Model	Acc@1	Acc@3	Acc@5
Random guess	2.94	8.82	14.71
Random initialized	38.98	61.71	74.22
FT only	41.50	64.93	76.53
PT only	46.14	69.30	79.61
PT + FT	45.89	68.96	79.47

Table 5: Instrument probing results.

	Chore	d Root	Chord quality		
Model	Acc@1	Acc@3	Acc@1	Acc@3	
Random guess	8.33	25.00	11.11	33.33	
Random initialized	48.86	78.76	38.05	77.97	
FT only	50.09	80.23	39.26	79.03	
PT only	62.93	89.42	50.09	85.58	
PT + FT	58.05	85.93	44.92	82.48	

Table 6: Chord probing results.

ing better instrument control (higher I-IOU), higher track-level similarity, both on pitch and groove, and more accurate duration prediction. Additionally, the results also show the neccesity of pre-training in this workflow. If removing it from training (no PT), the model can only achieve comparable performance on segment-level similarity metrics, while there are drops in all other metrics.

Last but not least, we conducted a comparative study between the proposed REMI-z and REMI+ (von Rütte et al. 2023), by training a randomly initialized model using REMI+ tokenization on this task. Compared to the No PT (REMI-z) model, using REMI+ resulted in less effective instrument control (indicated by lower Instrument IoU) and a decreased ability to understand instrument roles (indicated by higher Voice WER). Additionally, performance across all other metrics showed a notable decline. This suggests that in scenarios where track-wise music modeling is crucial or where effective instrument control is needed, the proposed REMI-z serves as a better alternative to REMI+ for multitrack music representation.

#### **Probing Analysis of Pre-Training Impact**

We have shown in the previous section that the proposed models outperform the baseline models that do not undergo the pre-traning stage on the generation quality. In this section, we further analyze the reason. Specifically, we conduct probing experiments to assess whether pre-training enhances the acquisition of musical knowledge during finetuning to facilitate understanding content conditions in finetuning.

We focus on two probing tasks: (1) classifying instrument types from content sequences that contain only position, pitch, and duration tokens, and (2) recognizing chord progression sequences from content sequences. To determine whether instrument or chord information is linearly accessible within the model's sequence embeddings, we employ linear classifier probes. In these probing experiments, the average pooling of the Transformer's output embeddings across all tokens in a sequence is fed into the classifier. The model parameters are kept frozen, and only the linear

classifiers are trained. We compare the knowledge captured by different models: a randomly initialized Transformer, a model that has undergone pre-training only (PT only), a model trained on the arrangement task without pre-training (FT only), and a model that has undergone both pre-training and fine-tuning (PT + FT).

**Instrument Type Probing** In this task, we use a linear probe to estimate the instrument type from a single-track music sequence without providing instrument tokens. The goal is to predict which instrument is most likely to play the given note sequence. The performance is evaluated using top-1, top-3, and top-5 prediction accuracy metrics.

As shown in Table 5, a model initialized with random weights shows notable improvement after fine-tuning, suggesting that the ability to discern instrument styles is the requirement for performing well in music arrangement tasks. However, the gains in accuracy are modest, likely due to the limited number of training samples, which may constrain the model's ability to aquire such knowledge through fine-tuning alone. Interestingly, models that underwent pretraining exhibit the highest accuracy in predicting instrument types. This indicates that substantial knowledge of instrument styles can be acquired effectively during the pretraining phase. Moreover, the proposed models that are both pre-trained and fine-tuned (PT+FT) maintained high accuracy levels, demonstrating that the knowledge about musical instruments is retained through the subsequent fine-tuning.

Chord Progression Probing In this task, we use linear probes to predict chord progressions from a 2-bar music content sequence without instrument tokens. Eight linear probes are trained simultaneously to predict the chord roots and qualities for a total of four chords (two chords per bar). This tasks is used to evaluate whether the model contains position-specific chord information. Similarly, we use top-1 and top-3 prediction acuracy metrics.

As shown in Table 6, the model with only fine-tuning (FT only) indicates a foundational grasp of chord knowledge for this complex music analysis tasks. However, similar to instrument prediction, the knowledge of recognizing chord progression does not significant gain, until the pre-training is also introduced into the model, confirming that pre-training establishes a robust basis for better understanding of content sequence, which may potentially help with the quality of arrangement.

#### Conclusion

We have demonstrated a method for transforming a music generation model, trained with next-token prediction, into a music arrangement model through unsupervised training. Specifically, we proposed a sequence-to-sequence training objective that is widely applicable across various arrangement scenarios. Under this training scheme, the instrument-specific style is derived from the provided music content and instrument constraints, resulting in high-fidelity arrangements. We also showed that small-scale fine-tuning is both effective and flexible for introducing various musical constraints, including instruments, voices, history, and track textures. Finally, we demonstrated that generative pre-training

not only enhances the empirical competence of the arrangement model but also helps formulate important musical concepts, facilitating fine-tuning for various downstream tasks.

#### References

- Alain, G.; and Bengio, Y. 2016. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*.
- Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language Models are Few-Shot Learners. *arXiv preprint arXiv:2005.14165*.
- Choi, K.; Hawthorne, C.; Simon, I.; Dinculescu, M.; and Engel, J. 2020. Encoding musical style with transformer autoencoders. In *International conference on machine learning*, 1899–1908, PMLR.
- Copet, J.; Kreuk, F.; Gat, I.; Remez, T.; Kant, D.; Synnaeve, G.; Adi, Y.; and Défossez, A. 2024. Simple and controllable music generation. *Advances in Neural Information Processing Systems*, 36.
- Crestel, L.; and Esling, P. 2016. Live Orchestral Piano, a system for real-time orchestral music generation. *arXiv preprint arXiv:1609.01203*.
- Dong, H.-W.; Donahue, C.; Berg-Kirkpatrick, T.; and McAuley, J. 2021. Towards automatic instrumentation by learning to separate parts in symbolic multitrack music. *arXiv preprint arXiv:2107.05916*.
- Huron, D. 2016. *Voice leading: The science behind a musical art.* mit Press.
- Ji, S.; Yang, X.; and Luo, J. 2023. A survey on deep learning for symbolic music generation: Representations, algorithms, evaluations, and challenges. *ACM Computing Surveys*, 56(1): 1–39.
- Kikuchi, Y.; Neubig, G.; Sasano, R.; Takamura, H.; and Okumura, M. 2016. Controlling output length in neural encoder-decoders. *arXiv preprint arXiv:1609.09552*.
- Lev, A. 2024. Los Angeles MIDI Dataset: SOTA kilo-scale MIDI dataset for MIR and Music AI purposes. In *GitHub*.
- Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; and Neubig, G. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9): 1–35.
- Loshchilov, I.; Hutter, F.; et al. 2017. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5.
- Lu, P.; Xu, X.; Kang, C.; Yu, B.; Xing, C.; Tan, X.; and Bian, J. 2023. Musecoco: Generating symbolic music from text. *arXiv* preprint arXiv:2306.00110.
- Madaghiele, V.; Lisena, P.; and Troncy, R. 2021. MINGUS: Melodic Improvisation Neural Generator Using Seq2Seq. In *IS-MIR*, 412–419.
- Makris, D.; Agres, K. R.; and Herremans, D. 2021. Generating lead sheets with affect: A novel conditional seq2seq framework. In 2021 International Joint Conference on Neural Networks (IJCNN), 1–8. IEEE.
- Malandro, M. E. 2023. Composer's Assistant: An Interactive Transformer for Multi-Track MIDI Infilling. *arXiv preprint arXiv:2301.12525*.
- Manilow, E.; Wichern, G.; Seetharaman, P.; and Le Roux, J. 2019. Cutting music source separation some Slakh: A dataset to study the impact of training data quality and quantity. In 2019 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), 45–49. IEEE.

- Min, L.; Jiang, J.; Xia, G.; and Zhao, J. 2023. Polyffusion: A diffusion model for polyphonic score generation with internal and external controls. *arXiv preprint arXiv:2307.10304*.
- Ou, L.; Ma, X.; Kan, M.-Y.; and Wang, Y. 2023. Songs across borders: Singable and controllable neural lyric translation. *arXiv* preprint arXiv:2305.16816.
- Qu, X.; Bai, Y.; Ma, Y.; Zhou, Z.; Lo, K. M.; Liu, J.; Yuan, R.; Min, L.; Liu, X.; Zhang, T.; et al. 2024. Mupt: A generative symbolic music pretrained transformer. *arXiv* preprint *arXiv*:2404.06393.
- Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I.; et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9.
- Ren, Y.; He, J.; Tan, X.; Qin, T.; Zhao, Z.; and Liu, T.-Y. 2020. Popmag: Pop music accompaniment generation. In *Proceedings of the 28th ACM international conference on multimedia*, 1198–1206.
- Rogers, A.; Kovaleva, O.; and Rumshisky, A. 2020. A Primer in BERTology: What We Know About How BERT Works. *Transactions of the Association for Computational Linguistics*, 8: 842–866.
- Scheffe, H. 1999. *The analysis of variance*, volume 72. John Wiley & Sons.
- Sennrich, R.; Haddow, B.; and Birch, A. 2015. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.
- Simon, I.; Morris, D.; and Basu, S. 2008. MySong: automatic accompaniment generation for vocal melodies. In *Proceedings of the SIGCHI conference on human factors in computing systems*, 725–734.
- Terao, M.; Hiramatsu, Y.; Ishizuka, R.; Wu, Y.; and Yoshii, K. 2022. Difficulty-aware neural band-to-piano score arrangement based on note-and statistic-level criteria. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 196–200. IEEE.
- Tiedemann, J.; and Scherrer, Y. 2017. Neural machine translation with extended context. *arXiv preprint arXiv:1708.05943*.
- von Rütte, D.; Biggio, L.; Kilcher, Y.; and Hofmann, T. 2023. FI-GARO: Controllable music generation using learned and expert features. In *The Eleventh International Conference on Learning Representations*.
- Wang, Z.; Min, L.; and Xia, G. 2024. Whole-song hierarchical generation of symbolic music using cascaded diffusion models. *arXiv* preprint arXiv:2405.09901.
- Wang, Z.; Wang, D.; Zhang, Y.; and Xia, G. 2020. Learning interpretable representation for controllable polyphonic music generation. *arXiv preprint arXiv:2008.07122*.
- Wei, S.; Xia, G.; Zhang, Y.; Lin, L.; and Gao, W. 2022. Music phrase inpainting using long-term representation and contrastive loss. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 186–190. IEEE.
- Yang, R.; Wang, D.; Wang, Z.; Chen, T.; Jiang, J.; and Xia, G. 2019. Deep music analogy via latent representation disentanglement. *arXiv* preprint arXiv:1906.03626.
- Yi, L.; Hu, H.; Zhao, J.; and Xia, G. 2022. Accomontage2: A complete harmonization and accompaniment arrangement system. *arXiv preprint arXiv:2209.00353*.
- Zhang, X.; Zhang, J.; Qiu, Y.; Wang, L.; and Zhou, J. 2022. Structure-enhanced pop music generation via harmony-aware learning. In *Proceedings of the 30th ACM International Conference on Multimedia*, 1204–1213.

Token	Meaning	X's range
i-X	Instrument type	0~128
o-X	Note's within-bar position	$0 \sim 127$
p-X	Note's pitch	$0 \sim 255$
d-X	Note's duration	$0 \sim 127$
b-1	Bar line	-
s-X	Time signature	$0 \sim 253$
t-X	Tempo	$0{\sim}48$

Table 7: REMI-z vocabulary

Zhao, J.; and Xia, G. 2021. Accommontage: Accompaniment arrangement via phrase selection and style transfer. *arXiv* preprint *arXiv*:2108.11213.

Zhao, J.; Xia, G.; and Wang, Y. 2023a. AccoMontage-3: Full-Band Accompaniment Arrangement via Sequential Style Transfer and Multi-Track Function Prior. *arXiv preprint arXiv:2310.16334*.

Zhao, J.; Xia, G.; and Wang, Y. 2023b. Q&A: Query-based representation learning for multi-track symbolic music re-arrangement. *arXiv preprint arXiv:2306.01635*.

#### **REMI-z Tokenization**

#### Vocabulary

We present the full vocabulary of the proposed REMI-z tokenizer in Table 7, along with the corresponding value range for each token type. Specifically, pitch token values correspond to MIDI pitch numbers, and instrument token values align with MIDI program numbers. Both position and duration tokens are measured in units of a 48th note (one-third of a sixteenth note). For time signature and tempo tokens, the actual values are first quantized before being mapped to their respective tokens. The mapping between token values and the actual values they represent is somewhat complex due to this quantization; further details can be found in the code.

Time signature and tempo tokens were utilized during pre-training but were not used in our fine-tuning experiments. For fine-tuning, we selected only 4/4 time signature songs from the Slakh2100 dataset (Manilow et al. 2019), which constitute the vast majority of the dataset (94.79% of all songs). The time signature token was omitted because all resulting songs share the same time signature. Tempo tokens were also excluded since the tempo of a song can be easily adjusted in digital audio workstation software as a singlevalue attribute, with changes affecting only the tempo token without impacting the organization of other tokens. This omission operates under the assumption that the composing and playing styles remain consistent across different tempos. This assumption may not be valid for cases with significant tempo differences. Therefore, in future research, if modeling that accounts for time-signature or tempo-specific characteristics is required, these tokens should also be included during fine-tuning.

#### **Example**

We name the tokenization scheme *REMI-z* because it encodes musical notes within a bar in a 'zig-zag' manner, as

illustrated in Figure 2a. As shown, notes are first encoded track-by-track and then bar-by-bar. This approach groups notes belonging to the same instrument together, facilitating track modeling. The resulting REMI-z sequence is depicted in Figure 2b. In contrast, REMI+ (von Rütte et al. 2023) encodes music column by column, strictly sorting by the position of the notes. This method distributes notes of the same instrument sparsely throughout the sequence, making instrument modeling more challenging. However, the REMI+ order naturally encodes global content information, which is why we used this note order in the content sequence (without instrument tokens), as shown in Figure 3.

### **Objective Metrics**

Due to page limitations, we formally define the objective metrics used for the voice separation task here.

We begin by defining two operators, WER and IoU. The operator WER(seq $_{out}$ , seq $_{ref}$ ) calculates the word error rate (WER) between the output sequence seq $_{out}$  and the reference sequence seq $_{ref}$ , defined as

WER = 
$$\frac{S + D + I}{N}$$
,

where S, D, and I represent the number of substitutions, deletions, and insertions, respectively, in  $\operatorname{seq}_{\operatorname{out}}$  compared to  $\operatorname{seq}_{\operatorname{ref}}$ , and N is the length of the reference sequence. The operator  $\operatorname{IoU}(\operatorname{set}_1,\operatorname{set}_2)$ , or Jaccard index, calculates the intersection-over-union of two unordered sets, defined as

$$IoU(set_1, set_2) = \frac{|set_1 \cap set_2|}{|set_1 \cup set_2|}.$$

Additionally, we define the operator  $S(\cdot)$ , which constructs a set from the elements of a sequence:

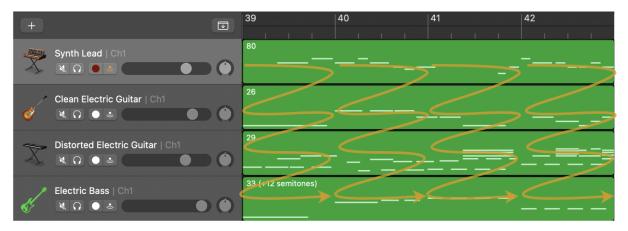
$$S(seq) = \{x \mid x \text{ is an element of seq}\}.$$

The operators  $I(\cdot)$  and  $C(\cdot)$  are defined as in §*Arrangement Fine-Tuning* of the paper.  $I(\cdot)$  extracts the instrument sequence, where elements are all instrument tokens appearing in the REMI-z sequence, ordered by their voice relationship.  $C(\cdot)$  extracts the content sequence from REMI-z. To facilitate the description of metrics, we introduce three additional operators:  $P(\cdot)$ ,  $O(\cdot)$ , and  $O(\cdot)$ , which extract the pitch sequence, position sequence, and duration sequence, respectively. Each sequence contains only the specified type of token, and the order of tokens remains consistent with that in the REMI-z sequence.

Below, we introduce each of the adopted metrics, along with their calculation equations and motivations. The equations define sample-wise metrics. The final performance is obtained by averaging the metric values across all output-reference pairs in the test set.

• Instrument IoU (I-IoU): This metric calculates the overlap of instruments between the output and target music, reflecting how accurately the generated music satisfies the desired instrument conditions. It is defined as:

$$IoU(S(I(seq_{out})), S(I(seq_{ref})))$$
.



(a) A 4-bar segment of music. The orange arrows indicate the 'zig-zag' order in which notes are encoded in REMI-z.

```
(s-9 t-35) (optional)
            i-80 o-18 p-74 d-14 o-36 p-76 d-11 i-26 o-0 p-60 d-26 i-29 o-0 p-36 d-10 o-12 p-36 d-12 o-18 p-48 d-12
1st bar
           _<mark>o-24 p-36 d-8 o-30 p-52 d-11 o-36 p-36 d-10 o-42 p-52 d-7 i-33</mark> o-0 p-36 d-23 b-1
            (s-9 t-35) (optional)
            i-80 o-0 p-74 d-10 o-12 p-76 d-5 o-18 p-74 d-10 o-30 p-72 d-5 o-36 p-71 d-5 o-42 p-72 d-10 i-26 o-0 p-67
2nd bar -
            d-13 o-18 p-67 d-12 o-30 p-67 d-11 o-42 p-64 d-11 i-29 o-6 p-43 d-7 o-12 p-31 d-6 o-18 p-47 d-12 o-24
            p-32 d-7 o-30 p-44 d-10 o-36 p-32 d-10 o-42 p-44 d-9 i-33 o-0 p-43 d-17 o-24 p-44 d-9 o-36 p-44 d-8 b-1
            (s-9 t-35) (optional)
            i-80 o-36 p-69 d-4 o-42 p-72 d-5 i-26 o-6 p-64 d-11 o-18 p-60 d-17 i-29 o-0 p-33 d-8 o-6 p-45 d-10 o-12
3rd bar -
            p-33 d-8 o-18 p-60 d-19 p-57 d-19 p-52 d-19 p-48 d-8 o-24 p-33 d-8 o-30 p-48 d-8 o-36 p-33 d-4 o-42
            p-48 d-4 i-33 o-0 p-45 d-8 o-12 p-45 d-8 o-24 p-45 d-8 o-36 p-45 d-8 b-1
            (s-9 t-35) (optional)
            i-80 o-0 p-76 d-5 o-6 p-74 d-10 o-18 p-74 d-5 o-24 p-72 d-5 o-30 p-72 d-4 o-36 p-74 d-5 o-42 p-72 d-13
            i-26 o-6 p-64 d-11 o-18 p-62 d-12 o-30 p-60 d-5 o-36 p-62 d-5 o-42 p-64 d-11 i-29 o-0 p-40 d-8 o-6 p-52
4th bar
            d-9 o-12 p-40 d-8 o-18 p-62 d-12 p-59 d-12 p-55 d-12 p-52 d-11 o-24 p-40 d-8 o-30 p-60 d-4 p-52 d-11
            o-36 p-62 d-5 p-40 d-8 o-42 p-60 d-13 p-57 d-13 p-53 d-13 p-48 d-8 i-33 o-0 p-40 d-8 o-12 p-40 d-8 o-24
            p-40 d-8 o-36 p-40 d-8 b-1
```

(b) The resulting REMI-z sequence from the segment above. Different instrument sequences (track sequences) are highlighted in different colors: Red for synth lead, blue for clean electric guitar, orange for distorted electric guitar, and purple for electric bass.

Figure 2: An example of REMI-z tokenization.

• Voice WER (V-WER): This metric compares the differences in voice relationships among instruments between the output and reference music, indicating how accurately the generated music meets the voice control condition. It is defined as:

$$WER(I(seq_{out}), I(seq_{ref})).$$

• Segment-level Pitch IoU: This metric measures the global similarity between the output and reference based on pitch information. A higher pitch overlap indicates greater tonal similarity, contributing to the faithfulness of the arrangement. Pitch IoU (P-IoU) between two REMIz sequences is calculated as:

$$\operatorname{P-IoU}(\operatorname{seq}_1,\operatorname{seq}_2) = \operatorname{IoU}(\operatorname{S}(\operatorname{P}(\operatorname{C}(\operatorname{seq}_1))),\ \operatorname{S}(\operatorname{P}(\operatorname{C}(\operatorname{seq}_2)))).$$

Segment-level P-IoU is then defined as:

$$P$$
-IoU(seq<sub>out</sub>, seq<sub>ref</sub>).

 Segment-level Position IoU: This metric assesses global similarity based on position information. Higher position overlap indicates greater groove similarity, enhancing the arrangement's faithfulness. Position IoU (O-IoU) between two REMI-z sequences is calculated as:

$$O-IoU(seq_1, seq_2) = IoU(S(O(C(seq_1))), S(O(C(seq_2)))).$$

Segment-level O-IoU is then defined as:

$$O-IoU(seq_{out}, seq_{ref}).$$

Melody Recall (M-R): This metric measures how well
the original melody of the song is preserved. For each
melody note in the reference, if there is a corresponding
note in the output with the same position and pitch, it
counts as a successful recall. It is defined as:

$$\frac{\left|\left\{\left(o,p\right)|\left(o,p\right)\in \mathcal{M}(\mathsf{seq}_{\mathsf{ref}})\wedge\left(o,p\right)\in \mathsf{seq}_{\mathsf{out}}\right\}\right|}{\left|\mathcal{M}(\mathsf{seq}_{\mathsf{ref}})\right|},$$

where  $M(\cdot)$  extracts the melody from REMI-z. In our experiment, this is implemented as a sequence of tuples (position, pitch) from the highest note at each position

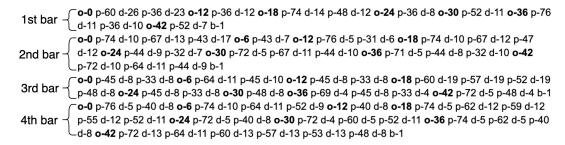


Figure 3: The resulting content sequence derived by the operator  $C(\cdot)$  from the REMI-z sequence shown in Figure 2b.

in the track with the highest average pitch. This assumes that the melody is always the highest note in the music, which is usually valid because the highest voice in a musical texture often carries the melody due to its perceptual prominence (Huron 2016).

• Track-level Pitch IoU: This metric applies P-IoU to track sequence pairs that correspond to the same instrument in both the output and reference sequences. It is defined as:

$$\frac{1}{|\mathbf{I}_o|} \sum_{i \in \mathbf{I}_o} \text{P-IoU}(\mathsf{seq}_\mathsf{out}[i], \mathsf{seq}_\mathsf{ref}[i]),$$

where  $I_o = S(I(seq_{ref})) \cap S(I(seq_{out}))$  and i refers to the instrument type. The sequence seq[i] extracts the track corresponding to instrument i from the REMI-z sequence.

• Track-level Position IoU: This metric applies O-IoU to track sequence pairs that correspond to the same instrument in both the output and reference sequences. It is defined as:

$$\frac{1}{|\mathcal{I}_o|} \sum_{i \in \mathcal{I}_o} \text{O-IoU}(\mathsf{seq}_\mathsf{out}[i], \mathsf{seq}_\mathsf{ref}[i]),$$

where i refers to the instrument type.

 Absolute Average Duration Difference (D-D): This metric measures the accuracy of duration prediction. It is defined as:

$$\frac{1}{|\mathbf{I}_o|} \sum_{i \in \mathbf{I}_o} \left| d_{\text{avg}}(\text{seq}_{\text{out}}) - d_{\text{avg}}(\text{seq}_{\text{ref}}) \right|,$$

where  $I_o = S(I(seq_{ref})) \cap S(I(seq_{out}))$ , and  $d_{avg}(seq)$  is the average duration of notes in seq, calculated as:

$$d_{\text{avg}}(\text{seq}) = \frac{1}{|\mathcal{D}(\text{seq}_{\text{ref}})|} \sum_{d \in \mathcal{S}(\mathcal{D}(\text{seq}))} d.$$

## **Subjective Metrics**

These metrics are based on listeners' auditory experiences and their subjective feelings while listening to the music. Since they are subjective, they cannot be easily defined using mathematical equations. However, we detail the evaluation criteria through the prompt questions provided in the questionnaire, which participants answered after listening to the demos. Each metric is assessed on a 5-point scale, ranging from 1 (very low) to 5 (very high).

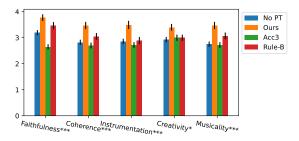
Hyper-parameter	Value
learning_rate	0.0005
train_batch_size	12
eval_batch_size	12
seed	42
gradient_accumulation_steps	8
total_train_batch_size	96
	Adam with
optimizer	betas=(0.9,0.999)
•	and epsilon=1e-08
lr_scheduler_type	cosine
lr_scheduler_warmup_steps	1000
num_epochs	1

Table 8: Pre-train hyper-parameter setting.

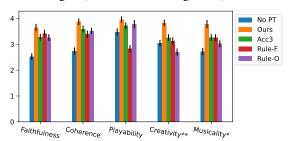
- Coherence: Does the arrangement flow naturally and smoothly? How consistent is each instrument's performance and style throughout the piece?
- Creativity: How creative is the arrangement while maintaining faithfulness and naturalness?
- **Musicality**: What is the overall musical quality?
- Faithfulness (band): How closely does the arrangement resemble the original piece in terms of melody and overall feel?
- **Faithfulness** (piano): How closely does the arrangement capture the overall feel of the original piece?
- **Instrumentation** (band-only): Does each instrument fulfill its appropriate role within the band, and do they harmonize effectively?
- **Playability** (piano-only): How well is the piece suited for piano? How likely is it that a human pianist could perform this accompaniment?
- **Compatibility** (drum-only): Is the drum beat compatible with the other instruments?
- Phrase Transition (drum-only): How effectively does the drum arrangement handle transitions between phrases?

#### **Subjective Evaluation Details**

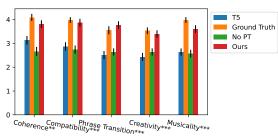
For the band arrangement task, we use an out-of-domain test with new compositions and new instrument groups for subjective evaluation, while for piano reduction and drum



(a) Error bars for the band arrangement task. Acc3 refers to AccoMontage-3 (Zhao, Xia, and Wang 2023a).



(b) Error bars for the piano reduction task.



(c) Error bars for the drum arrangement task. T5 refers to the Composers' Assistant model (Malandro 2023).

Figure 4: Error bar charts for the arrangement tasks.

arrangement, we use songs from the test set to facilitate comparison with the original piano track (Rule-O) and the ground truth drum track. Each model/method was tasked with arranging the full song, and a phrase from the chorus, the most representative part, was selected for comparison, each containing 15s–30s of audio. We prepared 6 songs for the band arrangement, 5 for the piano reduction, and 5 for the drum arrangement. In the questionnaire, participants were first presented with the original song clip to be arranged, followed by the anonymized results from different models/methods in a random order. Participants were asked to evaluate each arranged clip on a 5-point scale, using the questions provided for each metric as outlined in §Subjective Metrics in this Appendix. The group of songs evaluated by each participant (containing one original arrangement and all corresponding model outputs) was also randomly selected. In total, we received 56, 46, and 45 groups of feedback for the band arrangement, piano reduction, and drum arrangement tasks, respectively.

Here are specific details for the band arrangement setting. In the band arrangement tasks, we tested the models' ability in different scenarios by using various instrument settings. The content conditions were derived from either a piano arrangement or a full band arrangement, while the instrument condition included three different instrument sets: string trio (violin, viola, cello), rock band (synth lead, clean electric guitar, distorted electric guitar, electric bass), and jazz band (saxophone, violin, brass section, clean electric guitar, piano, string ensemble, electric bass). Among the 6 songs used for the band arrangement task, two songs were arranged in each of the three settings. Additionally, the input came from songs composed for different settings, including 3 piano solos and 3 band arrangements.

## **Hyperparameter Settings**

Below are the detailed hyperparameter settings used for the experiments.

For the pre-training experiment, we used a GPT-2 model with 12 layers of Transformer decoder blocks and a hidden size of 768. The training setup is summarized in Table 8. The hyperparameters were set without further tuning for optimal results. The experiment was conducted on a Linux platform using pytorch and transformers frameworks for pre-training. Fine-tuning was done with pytorch, transformers, and lightning.

For the fine-tuning experiments, we performed a simple search for the learning rate from the set {1e-5, 5e-5, 1e-4} and selected the best learning rate based on validation loss. The final learning rates were 5e-5 for drum arrangement and 1e-4 for band arrangement, piano reduction, and voice separation. Other hyperparameters were set as-is without further exploration. The batch sizes were set to 24, 24, 8, and 12 for band, piano, drum, and voice separation experiments, respectively, with corresponding context lengths of 768, 768, 1536, and 1024. We used the AdamW optimizer with a weight decay of 0.01, a linear learning rate scheduler, and a 500-step warmup. Training was conducted for 3 epochs for band, piano, and drum tasks, and 10 epochs for voice separation, with early stopping set to a 2-epoch patience. The best checkpoints were selected based on validation loss.

For the probing experiments, the batch size was set to 12 for chord probing and 64 for instrument probing. The learning rates were 5e-4 for chord probing and 1e-4 for instrument probing. Both experiments shared the same remaining hyperparameters: training for 10 epochs, using a linear learning rate scheduler, a 500-step warmup, and a weight decay of 0.01.

All experiments were run once with a fixed seed of 42.

#### **Arrangement Result Details**

Due to page limitations, we reported only the average scores for each metric and model in the main paper. As shown in Figure 4, we have plotted error bars for each comparison setting in the band arrangement, piano reduction, and drum arrangement tasks. The error bars include the mean score and the standard error of the mean

(SEM). Additionally, we conducted significance tests between our model and the major baselines in different tasks—AccoMontage-3 (Zhao, Xia, and Wang 2023a) for band arrangement and piano reduction, and Composers' Assistant (Malandro 2023) for drum arrangement—using withinsubject (repeated-measures) ANOVA (Scheffe 1999). The pvalues are indicated on the metric names in each plot, representing the significance levels: \* for p-values less than 0.05, \*\* for p-values less than 0.01, and \*\*\* for p-values less than 0.001.