IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 5, 2022, accepted February 19, 2022, date of publication February 28, 2022, date of current version March 17, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3155467

Translating Melody to Chord: Structured and Flexible Harmonization of Melody With Transformer

SEUNGYEON RHYU[1], **HYEONSEOK CHOI**[1], **SARAH KIM**[1], **AND KYOGU LEE**[1,2], **(Member, IEEE)**
[1]Department of Intelligence and Information, Seoul National University, Seoul 08826, Republic of Korea
[2]Artificial Intelligence Institute, Seoul National University, Seoul 08826, Republic of Korea

Corresponding author: Kyogu Lee (kglee@snu.ac.kr)

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by Institutional Review Board (IRB) of Seoul National University.

ABSTRACT Recent deep learning approaches for melody harmonization have achieved remarkable performance by overcoming the uneven chord distributions of music data. However, most of these approaches have not attempted to capture an original melodic structure and generate structured chord sequences with appropriate rhythms. Hence, we use a Transformer-based architecture that directly maps lower-level melody notes into a semantic higher-level chord sequence. In particular, we encode the binary piano roll of a melody into a note-based representation. Furthermore, we address the flexible generation of various chords with Transformer expanded with a VAE framework. We propose three Transformer-based melody harmonization models: 1) the standard Transformer-based model for the neural translation of a melody to chords (STHarm); 2) the variational Transformer-based model for learning the global representation of complete music (VTHarm); and 3) the regularized variational Transformer-based model for the controllable generation of chords (rVTHarm). Experimental results demonstrate that the proposed models generate more structured, diverse chord sequences than LSTM-based models.

INDEX TERMS Music information retrieval, computer generated music, neural networks, self-supervised learning.

I. INTRODUCTION

Automatic melody harmonization, which finds a coherent chord sequence that fits the given notes in a melody, is an essential topic in music generation. This task, which imitates the harmonizing process, is important for understanding human composition [1]. It is also practical for commercial use since it can reduce barriers to creating music without expertise [2], [3].

A melody harmonization task requires capturing the long-term dependencies in music since a constrained sets of chord progressions can consistently interact with a given melody [4]. This has motivated the use of linguistic techniques such as context-free grammar [5], genetic algorithms [6], or hidden Markov models (HMMs) [3], [7], [8]. Recently, deep learning approaches with bidirectional long short-term memory (BLSTM) showed robust performance by effective nonlinear sequential modeling of bar- or half-bar-based melody and chords [9]–[11]. Moreover, these studies successfully overcame the uneven chord distributions that are in common musical data.

Nevertheless, these LSTM-based studies had limitations in generating concrete chord structures. First, the models were unable to encode an original melodic structure despite their sequential architectures [4]. The notes in a melody were aggregated within a chord duration into a pitch-class histogram before being fed to the model. Second, the models did not explicitly consider capturing the patterns of chord progressions. Chord labels correspond to the constant time grids (e.g., a bar or half-bar). Sequential modeling of grid-based chord labels is likely to result in ambiguous patterns or hierarchies of the generated outputs [8].

The associate editor coordinating the review of this manuscript and approving it for publication was Juntao Fei.

Hence, we attempt to utilize a recent language model, Transformer, for structured melody harmonization. Transformer directly encodes inter- and intra-structures between two sequential data in dynamic length [12]. Thus, with Transformer, we can approach melody harmonization as the *translation* between two different languages, melody notes and chord labels, which share a semantic musical context.

However, conventional Transformer-based studies encoded music as a series of musical events [15]. Using event-based representations differs from how humans perceive a rendered or score-written melody for harmonization [16]. Instead, a grid-based melody representation can be more intuitive for modeling melodic patterns synchronized with chord labels [4], [17], [18]. In our work, we convert a melody into a more intuitive *note-based* representation, where each frame represents one note. To this end, we use a novel *time-to-note* compression method to map a binary piano roll representation into a note-based embedding.

In addition, we expand the conventional chord prediction task to a *flexible* harmonization task using a variational autoencoder (VAE) [19]. A melody can introduce diverse interpretations from multiple perspectives toward its musical structure or the arrangers' personalities [10], [20]. Therefore, it is more intuitive to *sample* chords from the proper distribution of real-world music. Current music generation systems have also leveraged VAE-based methods to produce creative outputs from the latent space [21]. However, most previous studies of melody harmonization have aimed at the static generation of chords with fixed model parameters. Thus, we utilize the VAE setting, which explicitly approximates the general chord distribution, for stochastic harmonization.

We concretely use the *variational Transformer* inspired by Lin *et al.* [22]. They used a Transformer-based model extended by a conditional VAE framework to generate a *response* from a conditional *context*. We leverage this seq2seq architecture to achieve a variational neural machine translation (VNMT) from a given melody to the chords [23]–[25]. To the best of our knowledge, we are the first to apply the VNMT approach to music generation. In particular, our approach is different from previous music generation studies using the variational Transformer, which mostly served as an *autoencoder* [26], [27].

Furthermore, we attempt to regularize the variational Transformer for *controlling* the chord outputs through a disentangled representation. Generating arbitrary sets of chords may not satisfy users who would like to create music based on their own tastes. In terms of building interactive music generation systems as well as learning a good representation for sequential data, controllable generation with the VAE framework has mainly been approached by recent studies. These studies have aimed to learn disentangled representations for high-level musical features, such as pitch, rhythm, harmony, context, or arousal, through supervised learning [28]–[31]. Inspired by these studies, we use domain-specific inductive bias to achieve a disentangled representation for the well-summarized context of the target melody and chords. In particular, we exploit an auxiliary regularization method proposed by Pati and Lerch [32] to force the target representation to be related to the musical attribute. We set the number of unique chords in a chord progression as a controllable attribute of the generated chords.

In this paper, we propose *three* Transformer-based models for structured and flexible chord generation from a given melody. These models are based on three types of Transformer architecture: 1) the **S**tandard **T**ransformer for structured **Harm**onization (**STHarm**), 2) the **V**ariational **T**ransformer for flexible **Harm**onization (**VTHarm**), and 3) the **r**egularized **V**ariational **T**ransformer for controllable **Harm**onization (**rVTHarm**). Our contribution also lies in the substantial evaluations of each model's performance using multiple datasets. One dataset is a benchmark dataset of popular music that is used for the direct comparison with previous approaches. The other dataset contains music from the contemporary genre, such as jazz, which possesses relatively higher musical tension than popular music. These datasets also differ by whether a key signature is normalized. Therefore, we assess the harmonization models in various dataset settings. The experimental results support that STHarm, VTHarm, and rVTHarm can capture structured contexts within and between melody and chord sequences, increase chord diversity, and explicitly control chord outputs, respectively, compared to LSTM-based models. The source code for the proposed methods is available at https://github.com/rsy1026/harmonizers_transformer.

II. RELATED WORKS

A. MELODY HARMONIZATION

Rule-based studies aim to simulate structural chord progressions carefully using linguistic techniques and heavy domain knowledge [4], [5], [33], [34]. Generic algorithms (GAs) were early probabilistic solutions that were combined with rule-based constraints [6], [20]. Machine learning approaches such as hidden Markov models (HMMs) demonstrate the use of probabilistic modeling to assess temporal dependency in music [3]. However, due to the inability of a standard HMM to capture elaborate harmonic functions, the HMM-based model was improved with domain knowledge [7] or tree-structured Markov models based on probabilistic context-free grammar [8], [35].

Lim *et al.* [9] utilized a stacked bidirectional long short-term memory (BLSTM) model to predict a chord for each bar of a given melody that was aggregated into a pitch-class histogram. This LSTM-based approach successfully improved model robustness for the skewed distribution of commonly used chords. Recently, Yeh *et al.* [10] revisited a sufficient number of conventional methods and consequently proposed MTHarmonizer, a deep multitask model that predicts chords with correct phrasings by directly supervising harmonic functions. Canonical metrics for assessing the coherence and diversity of the created chord sequence were also proposed. Sun *et al.* [11] used the orderless neural autoregressive

TABLE 1. Summary of the differences between the previous and proposed approaches for melody harmonization. The top attributes are related to the model architecture and experimental settings. The bottom attributes are related to the objectives of the studies. Bolded text indicates distinct differences between the proposed methods and the other methods.

Model	Lim *et al.* [9]	Yeh *et al.* [10]	Sun *et al.* [11]	**STHarm**	**VTHarm**	**rVTHarm**
Backbone	LSTM	LSTM	LSTM	**Transformer**		
Chord Unit	one bar	half-bar	half-bar	half-bar		
Dataset	Wikifonia.org [9]	HLSD [13]	HLSD	HLSD & **CMD [14]**		
Key Signature	Normalized	Normalized	Normalized	Normalized & **Not Normalized**		
Chord Diversity	Yes	Yes	Yes	No	Yes	Yes
Structuredness	No	Yes	No	Yes	Yes	Yes
Stochasticity	No	No	No	No	**Yes**	**Yes**
Controllability	No	No	No	No	No	**Yes**

distribution estimation (NADE) and the blocked Gibbs sampling method to approximate the complex joint probability among chords given a melody. They provided the model with a masked chord sequence so that the model could predict masked entries and leveraged class weights to efficiently balance the uneven distribution of chords.

These LSTM-based models shared the same data representation and model architecture. In particular, the models by Yeh *et al.* and Sun *et al.*, which improved the musical grammar or the diversity of chord types, were extensions of the model by Lim *et al.*. However, we assume that the LSTM-based approach is limited to modeling a serialized chord sequence without capturing the realistic pattern of chords. Our proposed models, which investigate the intrapatterns and interrelationship of the melody and chords, reveal the main difference in the model architecture. Table 1 summarizes additional details on how the proposed models differ from the LSTM-based approaches in terms of experimental settings and objectives.

B. TRANSFORMER-BASED MUSIC GENERATION

Music Transformer, introduced by Huang *et al.* [15], was one of the successful models for various objectives of long-term symbolic music generation. These researchers applied an event-based representation to polyphonic music performance data [36]. LakhNES used the extended Transformer architecture, Transformer-XL, to generate plausible multi-instrumental game sound chips [37]. Pop Music Transformer also used Transformer-XL and a novel data representation called ''revamped MIDI-derived events (REMI)'' was proposed to consider the metrical structure for generating polyphonic pop music [17]. Jazz Transformer adapted REMI to jazz music to create long-term coherent jazz lead sheets [38]. More recently, chord conditioned melody transformer (CMT) leveraged Transformer decoders to generate a *grid-based* melody given a chord progression [18]. This work attempted to create a melody with proper rhythms that were well aligned with the given chords. This work was similar to the current interest of our study.

Furthermore, Choi *et al.* [26] proposed a Transformer-based autoencoder that achieved global representation for the musical contexts of polyphonic piano performance data. Jiang *et al.* [27] introduced a hierarchical Transformer VAE to learn context-sensitive melody representation with self-attention blocks, enabling the model to control the melodic and rhythmic contexts.

C. MUSIC GENERATION FROM DISCRIMINATIVE LEARNING

Discriminative learning frameworks have been adapted to music generation studies. For example, Pati and Lerch [32] proposed a novel loss function for regularizing a latent variable to correlate with one musical attribute. EC^2-VAE and ExtRes decoupled representations of a melody's pitch and rhythmic attributes by intermediate supervision [28], [29]. FaderNet controlled polyphonic music by both low-level and high-level musical attributes using direct supervision and the regularization scheme from Pati and Lerch [30], [32]. Wang *et al.* [39] decoupled chord and texture attributes for interpretable generation of polyphonic music. PianoTree VAE aimed to achieve a representation of a tree-structured musical syntax [31].

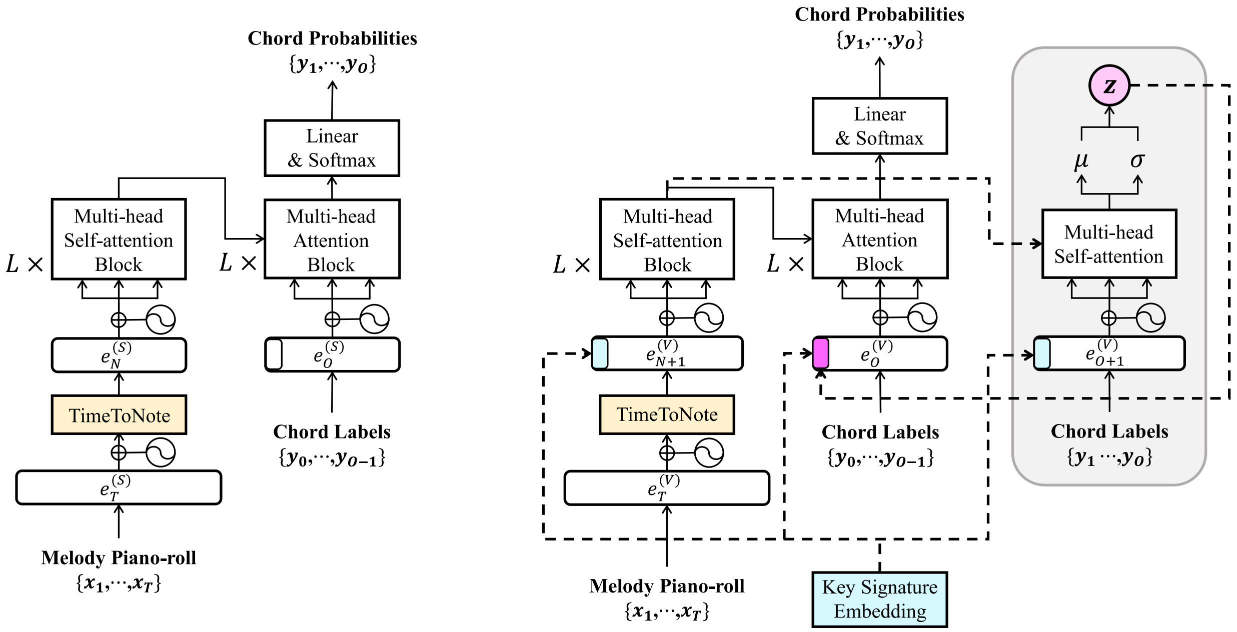
III. PROPOSED METHOD

We propose three models based on Transformer targeting structured and flexible melody harmonization. The first model uses the standard Transformer model to translate a melody to a chord sequence. The second model uses the variational Transformer to learn a global latent representation of the complete music [22]. The last model regularizes the representation of the variational Transformer to control harmonic attributes. We name these models *STHarm*, *VTHarm*, and *rVTHarm*, respectively. In each model, the Transformer encoder receives a given melody, and the decoder generates a chord sequence according to the attention weights computed between the melody and chords. The overall structures of the proposed models are illustrated in Fig. 1.

A. STANDARD TRANSFORMER MODEL (STHarm)

STHarm generally follows the original Transformer model, except that the input and output representations are not event-based [12]. Instead, we use a binary melody piano roll and serialized chord labels instead of musical event tokens. Each frame of the melody piano roll represents the same temporal length.

Let $x_{1:T} \in \{0, 1\}^{T \times |P|}$ be a one-hot vector sequence of a given melody, where T is the length of the melody, $|P|$ is the number of pitches, and t is a time index by the length of a



(a)

(b)

FIGURE 1. The overall architectures of the proposed methods: (a) STHarm and (b) VTHarm and rVTHarm. VTHarm and rVTHarm share the same architecture. The colored area and dotted lines represent the modified parts, from the vanilla Transformer.

sixteenth note. The encoder receives the input $x_{1:T}$ to capture the notewise melodic context as (1):

$$\begin{aligned} e_T^{(\mathrm{S})} &= \mathrm{Embedding}(x_{1:T}) \\ e_N^{(\mathrm{S})} &= \mathrm{TimeToNote}(e_T^{(\mathrm{S})} + w_T, M) \\ \mathrm{Enc}(x_{1:T}) &= \text{Self-AttBlocks}(e_N^{(\mathrm{S})} + w_N) \end{aligned} \tag{1}$$

where e_T, e_N, S, and N denote the time-level embedding vectors, note-level embedding vectors, STHarm, and the number of melody notes, respectively, Embedding and Self-AttBlocks denote the embedding layer and L multi-head self-attention blocks that are identical to the vanilla Transformer, respectively [12], w_* denotes a sinusoidal positional embedding scaled by a trainable weight [40], and TimeToNote is a novel method that we propose to convert the *timewise* embedding to the *notewise* embedding to capture the note patterns in a melody.

In the Time2Note procedure, we add the scaled positional embedding w_T to $e_T^{(\mathrm{S})}$. Then, we transfer it to the notewise embedding $e_N^{(\mathrm{S})}$ with average pooling by an alignment matrix $M \in \{0, 1\}^{T \times N}$ as (2), where M indicates the alignment path between a piano roll and a series of notes. This process enables each frame of the notewise embedding to preserve the information of the original note duration:

$$\mathrm{TimeToNote}(e, M) = \mathrm{Linear}\left(\frac{M^{\mathrm{T}} \cdot e}{\sum_{t=1}^{T} M_{t,1:N}}\right) \tag{2}$$

where Linear denotes a fully connected layer. The compressed embedding $e_{1:N}^{(\mathrm{S})}$ is added to another scaled positional embedding w_N and passes through the L multihead self-attention blocks.

The decoder receives the right-shifted target chords and computes attention with the encoder output $\mathrm{Enc}(x_{1:T})$ to predict the chords as (3):

$$\begin{aligned} e_O^{(\mathrm{S})} &= \mathrm{Embedding}(y_{0:O-1}) \\ \acute{e}_O^{(\mathrm{S})} &= \mathrm{AttBlocks}(e_O^{(\mathrm{S})} + w_O, \mathrm{Enc}(x_{1:T})) \\ p(\tilde{y}_{1:O}) &= \mathrm{Softmax}(\mathrm{Linear}(\acute{e}_{1:O}^{(\mathrm{S})})) \end{aligned} \tag{3}$$

where $y_{0:O-1} \in \{0, 1\}^{O \times |C|}$ is a sequence of one-hot vectors for the right-shifted target chords, O is the length of the chord sequence, $|C|$ is the number of chord classes, and AttBlocks denotes L loops of the Transformer attention blocks. The final probabilities are estimated by a final linear layer with softmax activation.

B. VARIATIONAL TRANSFORMER MODEL (VTHarm)

The proposed architecture of VTHarm is inspired by [22]. VTHarm has an additional probabilistic encoder for a latent variable z, where z represents the global attribute of the aggregated melody and chords. We denote this encoder as the *context encoder*. We add a global key signature label as a conditional input token to the model. The key signature is essential for an arbitrary melody to obtain a certain harmonic context [41]. The key signature token can aid the model in specifying the latent space and sampling the outputs from the constrained chord distributions. In contrast, STHarm does not use this token since it finds the mean distribution for chords that best fit a given melody.

The encoder used in VTHarm is identical to the encoder used in STHarm, except that the conditional token c is concatenated at the beginning of the note-based melody embedding $e_N^{(\mathrm{V})}$ as in (4):

$$\begin{aligned} e_{N+1}^{(\mathrm{V})} &= \mathrm{Concat}_s(c, e_N^{(\mathrm{V})}) \\ \mathrm{Enc}(c, x_{1:T}) &= \text{Self-AttBlocks}(e_{N+1}^{(\mathrm{V})} + w_{N+1}) \end{aligned} \tag{4}$$

where Concat_s denotes the concatenation over the sequence dimension. The self-attention block can connect c and the remaining parts of the embedding and convey any constraints to the whole embedding.

The context encoder infers the latent representation z from the encoder output, chord input y, and conditional token c as (5):

$$\begin{aligned} e_{O+1}^{(\mathrm{V})} &= \mathrm{Concat}_s(c, \mathrm{Embedding}(y_{1:O})) \\ \acute{e}_{O+1}^{(\mathrm{V})} &= \text{Self-AttBlock}(e_{O+1}^{(\mathrm{V})} + w_{O+1}) \\ r &= \mathrm{Concat}_d(\mathrm{Pool}(\mathrm{Enc}(c, x_{1:T})), \mathrm{Pool}(\acute{e}_{O+1}^{(\mathrm{V})})) \\ [\mu, \sigma] &= \mathrm{Linear}(r) \qquad z \sim \mathcal{N}(\mu, \sigma) \end{aligned} \tag{5}$$

where V denotes VTHarm, Concat_d denotes the concatenation over the feature dimension, Pool denotes the average pooling over time, and self-AttBlock denotes only one loop of the self-attention block. The context encoder maps the chord input $y_{1:O}$ into the embedding $e_O^{(\mathrm{V})}$. Then, c is concatenated at the beginning of $e_O^{(\mathrm{V})}$ over the sequence dimension before the multihead self-attention blocks. The self-attention output contains the harmonic context according to the key information. It is mean-aggregated over time so that it represents the global information of the chords [26]. The encoder output $E(c, x_{1:T})$ is also mean aggregated over time to represent the global attribute of a melody. These two aggregated vectors are concatenated over the feature dimension and pass through the bottleneck, resulting in two parameters, μ, and σ. The latent code z is inferred from μ and σ through the reparameterization trick, and its prior is assumed to be the normal distribution [19].

The decoder reconstructs the target chords from the right-shifted chord input and encoder output, conditioned by c and the latent variable z as (6):

$$\begin{aligned} e_o^{(\mathrm{V})} &= \mathrm{Concat}_s(z + c, \mathrm{Embedding}(y_{1:O-1})) \\ \acute{e}_O^{(\mathrm{V})} &= \mathrm{AttBlocks}(e_O^{(\mathrm{V})} + w_O, \mathrm{Enc}(x_{1:T})) \\ p(\tilde{y}_{1:O}) &= \mathrm{Softmax}(\mathrm{Linear}(\acute{e}_O^{(\mathrm{V})})) \end{aligned} \tag{6}$$

The right-shifted chord input is first encoded with the same lookup table from the context encoder. The latent variable z and the key signature token c are added to the beginning, which corresponds to the "start-of-sequence" part of the chord embedding. The following attention network transfers the aggregated information from z and c to all frames of the embedding. The rest of the Transformer decoder reconstructs the target chords.

C. REGULARIZED VARIATIONAL TRANSFORMER MODEL (rVTHarm)

Training VTHarm alone cannot guarantee a disentangled representation of the desired aspect. Therefore, rVTHarm aims to achieve a disentangled representation to control the generated chord outputs. We use the auxiliary loss by Pati *et al.* [32] to directly supervise the latent representation z. In this study, we choose the number of unique chords in the progression, or *chord coverage*, as a naive attribute for the chord complexity [10].

The regularization function from Pati *et al.* assumes that the target dimension of the latent representation can be disentangled by its monotonic relationship with a specific attribute [32]. For example, the target attribute value should increase when the constrained latent dimension is modulated toward a positive direction. To this end, the difference between the attribute values of an arbitrary pair of two samples is forced to be the same sign as that between the corresponding latent representations. Let a_i and a_j be the target attribute values of the ith and jth batches, respectively, where $i, j \in [1, B]$ and B is the batch size. Similarly, let z_i^r and z_j^r be the rth dimension values of the latent variables of the ith and jth batches, respectively. A distance matrix $\mathcal{D}_r$ is computed between all pairs of z_i^r and z_j^r in the mini-batch. The corresponding $\mathcal{D}_a$ is computed in the same way between all pairs of a_i and a_j. We minimize the difference between $\mathcal{D}_r$ and $\mathcal{D}_a$ as (7):

$$\mathcal{L}_{\mathrm{Reg}} = \mathrm{MSE}(\tanh(\mathcal{D}_r), \mathrm{sign}(\mathcal{D}_a)) \tag{7}$$

where MSE is the mean squared error. In this paper, we regularize the first dimension of z, so $r = 1$.

D. TRAINING OBJECTIVES

The main objective for STHarm is maximizing the log likelihood of the estimated chord sequence y given the melody x:

$$\mathcal{L}_{\mathrm{ST}} = \mathbb{E}[-\log p_\theta(y|x)] \tag{8}$$

where θ are the model parameters of STHarm.

In VTHarm, the main goal is to approximate the marginal distribution of y through the objective of negative evidence lower bound (ELBO) by minimizing the losses for the reconstruction and Kullback-Leibler divergence (KLD) [19]. The chord probability $p_\theta(y)$ and posterior distribution $q_\phi(z)$ are conditioned by the melody input x and key signature token c, whereas the prior $p_\theta(z)$ is the normal distribution following the conditional VAE framework [42]:

$$\begin{aligned} \mathcal{L}_{\mathrm{VT}} = \mathbb{E}_{q_\phi(z|x,y,c)}&[-\log p_\theta(y|x, z, c)] \\ &+ \lambda_{KL}\mathrm{KL}(q_\phi(z|x, y, c)\|p_\theta(z)) \end{aligned} \tag{9}$$

where q_ϕ is the posterior distribution of z parameterized by ϕ, and λ_{KL} is a hyperparameter for balancing the KLD loss term [21], [43].

This training objective is expanded in rVTHarm by the explicit regularization of the latent space. Therefore, rVTHarm shares the overall objective with VTHarm except for the added regularization term as (10):

$$\mathcal{L}_{\mathrm{rVT}} = \mathcal{L}_{\mathrm{VT}} + \lambda_{Reg}\mathcal{L}_{\mathrm{Reg}} \tag{10}$$

where λ_{Reg} is a hyperparameter for balancing the auxiliary loss term.

To generate chords, VTHarm and rVTHarm autoregressively sample the chord output $y_{1:O}$ from the melody input $x_{1:T}$, latent variable z, and conditional token c as (11):

$$p_\theta(y|x, z, c) = \prod_{O} p_\theta(y_o|x_{1:t}, y_{0:o-1}, z, c) \tag{11}$$

where z is sampled from the normal prior $\mathcal{N}(0, 1)$.

IV. EXPERIMENTAL SETTINGS

We conduct objective and subjective evaluations for the three proposed methods. In this section, we explain the settings for the corresponding experiments. We first introduce the two datasets used for the experiments. Next, we summarize the baseline models, model settings, and metrics for the evaluations.

A. DATASETS

We set $|P| = 13$ for the 12 pitch classes and rest. We convert all chords into one of the 72 chords, which are triad chords in major, minor, diminished, and seventh chords in major, minor, dominant, so that $|C| = 72$. Each note and chord are quantized by lengths of sixteenth note and a half-measure for all datasets, respectively. The length of each batch is a maximum of 8 measures. We only use songs with a time signature of 4/4 and all songs are set to 120 BPM. The training, validation, and test sets for each dataset are divided into approximately an 8:1:1 ratio. We construct batches by slicing each song into excerpts of 8-measures where 2-measures overlap. For each test, we extract 8-measure excerpts without an overlap. We use two public datasets that differ in some experimental settings as well as musical characteristics: the Chord Melody Dataset (CMD) and the Hooktheory Lead Sheet Dataset (HLSD).

1) THE CHORD MELODY DATASET (CMD)

CMD [14] is composed of 473 songs in contemporary genres such as jazz and pop. The songs in this database are only in the major key, and most of them are transposed to all 12 keys. We choose this dataset to examine the model performance from the complex chords in various keys with nontrivial tensions. The lead sheets are in the music extensible markup language (MusicXML) format, where the melody and chord labels are manually annotated and are parsed with the existing MusicXML parser [44], [45]. We use 389 songs for the training set and the rest for the validation and test sets (48 songs each). As a result, we use 36,528, 1,756, and 165 samples for the training, validation, and test sets, respectively.

2) THE HOOKTHEORY LEAD SHEET DATASET (HLSD)

HLSD [13] is an online database of melody and chord annotations that cover various genres, such as the pop, new age, and original soundtracks. This dataset has been constructed on a crowdsourcing platform called TheoryTab,[1] in which users have transcribed a large number of high quality melodies and chords. This dataset contains the raw annotations of melodies and chords in XML format, JSON data of the symbolic features of melodies and chords, and piano-roll figures depicting the melody and chords. We use the JSON data for 9,218 songs divided into 13,335 parts. We also normalize all songs into C major or C minor, as in previous studies [10], [11]. Following Sun *et al.* [11], we use 500 parts for the test set and the other 500 parts for the validation set. As a result, we use 32,619, 1,346, and 809 samples for the training, validation, and test sets, respectively.

B. COMPARATIVE METHODS

We use two baseline models and one ground truth for our study. **BLSTM** by Lim *et al.* [9] is composed of two stacked layers of bidirectional LSTM. This model has been a base for most of the recent deep learning approaches [10], [11]. We use BLSTM to compare the stacked RNN structure with Transformer. **ONADE** by Sun *et al.* [11] uses the orderless NADE and Gibbs sampling. This model represents a BLSTM-based model with randomness and improved chord diversity. For the ground truth, we use the original progressions from the datasets. We denote the ground truth as **Human**.

C. TRAINING

The embedding sizes of the melody and chord are 128 and 256, respectively. We use a hidden size of 256, attention head size of 4, number of attention blocks L of 4, and size of the latent variable z of 16. A dropout layer is used after every scaled positional encoding at a rate of 0.2. We use an Adam optimizer [46] with an initial learning rate of 1e-4, which is reduced to 95% after every epoch. We train the proposed models for 100 epochs with a batch size of 128. To select the value of λ_{KL}, we refer to several studies on VAE-based music generation in which a scaling weight smaller than 1 encourages better reconstruction [21], [47]. Then, we empirically set λ_{KL} and λ_{Reg} to be 0.1 and 1, respectively, which results in the best performance.

The models are implemented and evaluated in Python 3 and the PyTorch deep learning framework of version 1.5.0. For training each model, we use one NVIDIA GeForce GTX 1080 Ti. We mostly refer to the previous implementations [40], [48] when implementing the vanilla Transformer. For implementing and training BLSTM and ONADE, we use the original settings [9], [11]. The gradients are all clipped to 1 for the learning stability during training of all models. VTHarm, rVTHarm, and ONADE are assessed with 10 test samples per melody due to their randomness.

[1] https://www.hooktheory.com/theorytab

Other models are evaluated with the samples in maximum probabilities. We use the truncation trick with a threshold of 3 for VTHarm and rVTHarm in qualitative and subjective tests [49].

D. METRICS

We introduce three categories of metrics for evaluating the proposed models: chord coherence and diversity, harmonic similarity, and subjective evaluation.

1) CHORD COHERENCE AND DIVERSITY

We use six canonical metrics proposed by Yeh *et al.* that have been leveraged by recent studies [10], [11]. In brief, **chord histogram entropy (CHE)** and **chord coverage (CC)** measure chord diversity. **Chord tonal distance (CTD)** measures the coherence of the chord transition. **Chord tone to non-chord tone ratio (CTR)**, **pitch consonance score (PCS)**, and **melody-chord tonal distance (MTD)** measure the coherence between the melody and chords:

- **Chord histogram entropy (CHE).** This metric computes the entropy from the histogram of $|C|$ bins that counts the occurrences of the chord classes within the chord sequence:

$$\mathrm{CHE} = -\sum_{i=1}^{|C|} p_i \log p_i \tag{12}$$

where p_i denotes the probability of the ith bin of the histogram.

- **Chord coverage (CC).** This metric is the number of unique chord labels that occur in the chord sequence.
- **Chord tonal distance (CTD).** This metric is the Euclidean distance between the 6-D tonal feature vectors that represent the two adjacent chords. These vectors are calculated using the pitch class profile (PCP) features [50], [51]. We compute the average of the CTD values for all pairs of adjacent chords in each progression. Each CTD is calculated as (13):

$$\mathrm{CTD}_n(d) = \frac{1}{\|\mathbf{c}_n\|_1} \sum_{l=0}^{11} \Phi(d,l)\mathbf{c}_n(l) \qquad 0 \le d \le 5 \quad 0 \le l \le 11 \tag{13}$$

where n is the chord index, d is one of the dimension indices of the 6-D tonal space, c_n is the PCP vector of the nth chord, where the number of entries for the chord tones is 1 ($c_n \in \{0, 1\}$), l denotes one of the 12 entries of the PCP vectors, where each entry corresponds to each pitch class, and $\phi(d, l)$ denotes the dth basis of the 6-D tonal space for the lth entry of the PCP vector. Each basis is defined as (14):

$$\phi_l = \begin{bmatrix} \Phi(0,l) \\ \Phi(1,l) \\ \Phi(2,l) \\ \Phi(3,l) \\ \Phi(4,l) \\ \Phi(5,l) \end{bmatrix} = \begin{bmatrix} r_1 \sin l\frac{7\pi}{6} \\ r_1 \cos l\frac{7\pi}{6} \\ r_2 \sin l\frac{3\pi}{2} \\ r_2 \cos l\frac{3\pi}{2} \\ r_3 \sin l\frac{2\pi}{3} \\ r_3 \cos l\frac{2\pi}{3} \end{bmatrix} \qquad 0 \le l \le 11 \tag{14}$$

where ϕ_l is the complete transition matrix of the 6-D feature vector for the lth entry of the PCP vector, r_1, r_2 and r_3 are the radii of the three circles that represent the 6-D tonal space. They are set to 1, 1, and 0.5, respectively, as in Harte *et al.* [51].

- **Chord tone to non-chord tone ratio (CTR).** Originally named CTnCTR, this metric is the ratio of the number of chord tones compared to the number of nonchord tones and *proper* nonchord tones, which have a maximum of 2-semitone intervals to the right-after note:

$$\mathrm{CTR} = \frac{n_c + n_p}{n_c + n_n} \tag{15}$$

where n_c, n_n, and n_p denote the number of chord tones, nonchord tones, and proper nonchord tones, respectively, that are computed from the melody notes and corresponding chord labels.

- **Pitch consonance score (PCS).** This metric is a consonance score based on pitch intervals between the melody note and corresponding chord notes. The pitches of the melody notes are assumed to always be higher than those of the chord notes. According to the pitch interval, PCS is one of $\{-1, 0, 1\}$: 1 for perfect 1st and 5th, major/minor 3rd and 6th; 0 for perfect 4th; and -1 for other intervals. The PCS values within each sixteenth-note window are aggregated into the average. We compute the total average of the aggregated PCS for all windows over time.
- **Melody-chord tonal distance (MTD).** Originally named MCTD, this metric is the tonal distance between each melody note and its corresponding chord label. It is calculated in the same way as CTD. Each MTD value is weighted by the duration of the corresponding melody note. We average the MTD values for all of the melody notes and their chord labels.

2) HARMONIC SIMILARITY

We measure the similarity between the generated and human-composed chords with three metrics and assume that the chord progressions in the human-composed music inherit hierarchical and metrical structures [16], [52]. Hence, we set

TABLE 2. Evaluation results for chord coherence and diversity. CHE and CC measure the chord diversity, whereas the remaining four metrics measure the chord coherence: CTD measures the coherence of the chord progression itself. CTD, CTR, PCS, and MTD measure how harmonic the chord progression is with the given melody.

Dataset	Chord Melody Dataset						Hooktheory Lead Sheet Dataset					
Category	Diversity		Coherence				Diversity		Coherence			
Metric	CHE↑	CC↑	CTD↓	CTR↑	PCS↑	MTD↓	CHE↑	CC↑	CTD↓	CTR↑	PCS↑	MTD↓
BLSTM	1.380	5.297	0.497	1.170	**0.543**	**1.302**	0.928	3.262	0.609	1.146	**0.639**	**1.328**
ONADE	1.389	5.482	0.502	1.220	0.497	1.362	1.123	4.243	0.467	1.136	0.470	1.392
STHarm	1.349	5.030	**0.443**	1.213	0.428	1.396	0.994	3.193	**0.446**	**1.150**	0.522	1.396
VTHarm	**1.877**	**7.523**	0.631	1.225	0.374	1.428	**1.543**	**5.356**	0.696	1.147	0.459	1.435
rVTHarm	1.705	6.202	0.508	**1.227**	0.394	1.419	1.440	4.678	0.536	1.146	0.445	1.447
Human	1.618	6.412	0.580	1.301	0.389	1.408	1.356	4.686	0.626	1.180	0.497	1.400

the human-composed music as the ground truths of the structured harmonization. Concretely, a system that generates chord progressions *similar* to human-composed music is assumed to achieve more structured harmonization [35].

Briefly, **the Levenshtein edit distance (LD)** is the global matching score between two chord sequences. **The tonal pitch step distance (TPSD)** and **directed interval class distance (DICD)** measure the distance between two chord progressions:

- **Levenshtein edit distance (LD).** LD is the Levenshtein edit distance between the generated chord labels and the ground-truth labels [35]. It measures the extent to which the generated chords are substituted for human-composed chords.
- **Tonal pitch step distance (TPSD).** TPSD computes the geometrical dissimilarity between the generated chords and the ground-truth chords in terms of the tonal pitch space (TPS) chord distance rule [53]. The TPS between chord x and chord y is computed as (16):

  $$\mathrm{TPS}(x, y) = j + k \tag{16}$$

  where j is the least number of steps in one direction from the chordal root of x to that of y according to the circle-of-fifths rule. In the circle-of-fifths rule, all pitch classes are arranged in intervals of either perfect fifth or fourth [54]. The variable k is the number of unique pitch class indices in the four levels (root, fifths, triadic, diatonic) within the basic space of y compared to x [53]. That is, if the pitch class index is shared by y and x, it is not counted. We compute the TPS values between all pairs of adjacent chords within each progression, resulting in a step function. TPSD is calculated as the area between the two step functions derived from the two chord progressions.
- **Directed interval class distance (DICD).** DICD computes the city block distance between the directed interval class (DIC) representation vectors for the chord transitions [55]. DIC is the histogram vector of the directional pitch interval classes, ranging from −5 to 6, computed between all pairs of chord notes from the two adjacent chords. We calculate each pitch interval from each note of the first chord to all notes of the second chord. DICD indicates both the tonal distance and *direction* between the two successive chords.

3) SUBJECTIVE EVALUATION

We expand the conventional criteria [10], [11] for deeper analysis of human judgment. **Harmonicity** measures how coherent the chords are with a given melody. **Unexpectedness** measures how much the chords deviate from expectation. **Complexity** measures how complex chord progression is perceived to be. **Preference** measures personal favor for chord progression [9].

V. EVALUATION

In this section, we introduce the experimental results of the objective and subjective evaluations into several categories as follows. First, we compare the results of the proposed models in **chord coherence and diversity** with the baseline models. Next, we measure **harmonic similarity to human-composed music** for all models to examine whether the proposed models can result in structured harmonization. Then, we check with the **controllability of rVTHarm** for the intended factor compared with VTHarm. In addition, we introduce the results for the **subjective evaluation** and discuss the corresponding results. Moreover, we illustrate some **qualitative results** for all models to verify the strength of the proposed model. Last, we show an **ablation study** to investigate the influence of the information of the key signature added to the variational models.

A. CHORD COHERENCE AND DIVERSITY

We evaluate the overall coherence and diversity of the generated chords. Table 2 shows the results for all models. VTHarm and rVTHarm show higher CHE and CC than the baseline models in both datasets. This result indicates that these models have higher chord diversity than the baseline models. STHarm, on the other hand, reveals the lowest CTD and the lowest CHE and CC for all datasets except for CHE on HLSD. This implies that STHarm can generate smoother and simpler chord transitions than other models [11]. BLSTM and ONADE show better PCS and MTD but lower chord diversity than the proposed models.

Meanwhile, Human shows worse scores for chord coherence than STHarm for the following reasons. 1) The

TABLE 3. Evaluation results for the chord similarity metrics. Lower scores correspond to higher human composition similarity.

Dataset	Chord Melody Dataset		
Metric	LD↓	TPSD↓	DICD↓
BLSTM	**0.75(±0.20)**	2.63(±1.11)	116.45(±42.98)
ONADE	0.85(±0.17)	2.80(±1.06)	128.10(±41.51)
STHarm	0.80(±0.21)	**2.43(±1.35)**	**107.68(±44.86)**
VTHarm	0.86(±0.14)	2.72(±1.02)	121.08(±36.82)
rVTHarm	0.86(±0.15)	2.71(±1.17)	118.01(±36.59)
Dataset	Hooktheory Lead Sheet Dataset		
Metric	LD↓	TPSD↓	DICD↓
BLSTM	**0.62(±0.21)**	2.48(±1.11)	85.39(±35.22)
ONADE	0.90(±0.14)	2.75(±1.17)	116.16(±39.67)
STHarm	0.65(±0.25)	**2.17(±1.55)**	**75.54(±40.81)**
VTHarm	0.77(±0.16)	2.54(±1.15)	98.55(±33.53)
rVTHarm	0.79(±0.16)	2.32(±1.26)	91.63(±34.92)

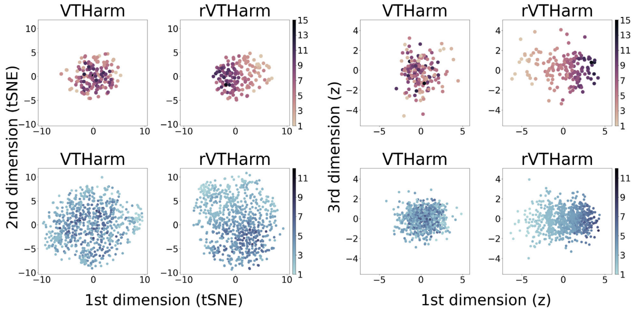
human-composed samples from CMD and HLSD include 72 different chord types with various amounts of musical tensions. 2) STHarm may generate common chords more frequently from the average chord distribution than the human-composed music, as shown in the lower diversity scores. Concretely, the most frequent chords in real-world music are diatonic chords such as the C, G, and F major chords in the C major key [9]. Since these chords have relatively less musical tension with respect to a melody, they are close to the melody under a music-theoretical space. Thus, these chords may obtain better coherence scores than other chords with more musical tension.

Moreover, Human shows lower diversity scores than the variational models. We assume that this is because these models can produce some infrequent chords far from the mean distribution of real-world music. The nature of stochastic generation models draws samples from the normal distribution [49]. Some of the generated chords may violate the given key signature but increase the information outside the certain harmonic context. Hence, they may contribute to higher chord diversity than human-composed music.

Consequently, the overall results reflect a trade-off between chord coherence and diversity [6], [10]. Additionally, Human cannot serve as the upper bound for the six metrics in both datasets. Therefore, these metrics cannot function as complete criteria for determining the *good* harmonization but only show the model tendencies in the music-theoretical perspective [10], [11]. Hence, we are inspired to use additional criteria to evaluate the generated outputs with respect to human-composed chords.

B. HARMONIC SIMILARITY TO HUMAN

We investigate the harmonic similarity between the human-composed and generated chords. We use the samples from Human as the ground truth. This explicit comparison with Human can provide insight into whether the generated chords from each model are as well-structured as human-composed music [8].



(a) (b)

FIGURE 2. Visualization of (a) tSNE results and (b) two dimension values from z. The top (purple) and bottom (indigo) rows represent the CMD and HLSD, respectively. The hue of each plot represents the chord coverage value.

TABLE 4. Pearson's correlation coefficients between α and CC of the generated outputs from VTHarm and rVTHarm. CMD and HLSD are the Chord Melody Dataset and Hooktheory Lead Sheet Dataset, respectively.

Dataset	CMD	HLSD
VTHarm	-0.1132	0.0805
rVTHarm	**0.5332**	**0.4512**

The harmonic similarity results are shown in Table 3. BLSTM shows the lowest LD compared to the proposed models, whereas ONADE shows the highest LD in all datasets. This indicates that BLSTM is better than the proposed models at providing the right chords to the melody. However, the better matching of individual chords does not correspond to the higher similarity of the chord sequence in terms of musical structure [53].

For TPSD and DICD, STHarm shows the lowest scores in all datasets. This implies that STHarm can generate chord patterns that is more similar to Human than other models. VTHarm and rVTHarm show higher LD scores than BLSTM but better similarity scores than ONADE. This indicates that the VT models tend to have higher substitution probabilities between chords than BLSTM [53]. This is possible because the VT models are trained to induce some infrequent chords that are far from the mean distribution of real-world chords. Nonetheless, the VT models are better than ONADE at creating more human-like chord patterns, even with a larger variety of chord types.

C. CONTROLLING CHORD COMPLEXITY

We verify the monotonic relationship between the chord attribute and z from rVTHarm. We use VTHarm and rVTHarm to infer z from the test melodies and chords. Then, the dimension of z is reduced by two with t-stochastic neighbor embedding (tSNE) [30]. When visualizing, we use the chord coverage value as the third dimension (hue). The tSNE results and two dimensions, the first and third, of the original z are illustrated in Fig. 2. This figure shows that the tSNE results of rVTHarm are grouped by the attribute compared

TABLE 5. Subjective evaluation results for the six methods according to whether the participants have known the given melody.

Condition	*With* Melody Awareness				*Without* Melody Awareness			
Metric	H	U	C	P	H	U	C	P
BLSTM	3.29(±1.00)	2.67(±1.04)	2.42(±0.98)	2.88(±1.11)	2.87(±1.05)	2.96(±1.02)	2.68(±0.95)	2.51(±1.10)
ONADE	2.91(±1.03)	2.98(±1.07)	2.89(±1.01)	2.69(±1.07)	2.76(±1.03)	3.09(±1.01)	2.90(±1.05)	2.57(±1.12)
STHarm	**3.44(±1.01)**	2.33(±0.99)	2.33(±1.08)	**3.11(±1.16)**	**3.20(±1.15)**	2.68(±1.00)	2.65(±1.01)	**2.92(±1.18)**
VTHarm	2.95(±1.04)	**3.23(±1.01)**	**3.05(±0.98)**	2.83(±1.07)	2.87(±1.11)	**3.18(±1.02)**	**2.97(±0.95)**	2.65(±1.06)
rVTHarm	3.18(±1.13)	2.95(±0.99)	2.89(±0.94)	2.98(±1.24)	2.81(±1.08)	3.06(±1.05)	2.87(±1.06)	2.51(±1.06)
Human	3.41(±1.13)	2.93(±1.05)	2.92(±1.00)	3.33(±1.17)	3.15(±1.15)	2.96(±1.04)	2.97(±1.08)	3.00(±1.19)

to VTHarm. The first dimension of z from rVTHarm is also shown to be monotonically related to the attribute [32].

In addition, we examine the attention maps of rVTHarm with different values of α. We randomly sample z, where α is set to be one of $\{-3, 0, 3\}$, and generate the chords from z and the test melodies. We sum the attention matrices along the head dimension to see the aggregated weights. Fig. 3 shows that the attention weights become balanced and diagonal when α increases from -3 to 3. This implies that the decoder of rVTHarm tends to focus on more melody notes when α increases.

Furthermore, we compute Pearson's correlation coefficients between α and the CC scores of the corresponding chord outputs. Table 4 shows that rVTHarm reveals higher correlation coefficients than VTHarm for all datasets. This confirms that rVTHarm derives a meaningful representation for the intended chord attribute compared to VTHarm.

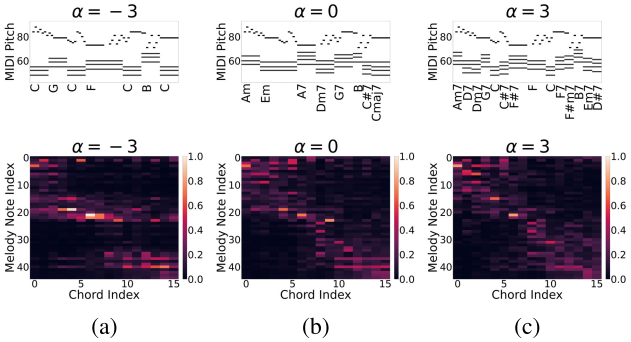


FIGURE 3. The generated results from rVTHarm in the piano-rolls (top) and the corresponding attention matrices (bottom). (a), (b), and (c) represent the results from different values of $\alpha \in \{-3, 0, 3\}$.

D. SUBJECTIVE EVALUATION

We conduct a listening test for subjective evaluation. We extract the samples in 8-measure length from the arbitrary parts of each melody. For rVTHarm, we sample z by setting a to randomly be $\{-3, 0, 3\}$. The listening test comprises ten trials, where each trial contains six samples of all comparative methods for one melody. A participant[2] grades four metrics, Harmonicity (H), Unexpectedness (U), Complexity (C), and Preference (P), on a five-point Likert scale for each method [10], [11]. We denote these metrics as "H", "U", "C", and "P" for simplicity. We collect answers on whether a participant is familiar with a given melody as in Lim *et al.* [9]. A total of 36 participants were involved in the listening test: 3 participants had degrees in music. Thirty-two participants indicated that they had musical backgrounds, and 25 participants mentioned that they usually listened to popular music.

Table 5 shows that the results mainly support the quantitative evaluation results. In contrast, STHarm shows the highest H score regardless of melody awareness. This suggests that STHarm outputs plausible chords to listen to than the baseline models. For U and C, VTHarm shows the highest scores, and the variational models show lower harmonicity and preference scores than STHarm. We assume that the variational models tend to generate more chords far from the mean distribution of the learned music data than STHarm. Such unique chords can reveal more inharmonicity than the frequent chords, and it may have provided the participants with unpleasant feelings. In addition, most participants listened to popular music, where common chords with less musical tension are used. Therefore, it may have led the participants providing poorer scores on preference as well as harmonicity. Nevertheless, VTHarm shows a better P score than ONADE with lower U and C scores. This means that VTHarm is more persuasive than the baseline model with lower chord complexity.

TABLE 6. Pearson's correlation coefficients of U score with P and C scores for Human (H), BLSTM (B), ONADE (O), STHarm (S), VTHarm (V), and rVTHarm (R) according to the melody awareness.

Model	H	B	O	S	V	R
P(aware)	-0.17	-0.09	-0.04	0.01	-0.08	**-0.30**
C(aware)	0.47	0.65	0.56	0.49	0.53	**0.29**
P(unaware)	-0.21	-0.22	-0.03	-0.10	-0.16	-0.06
C(unaware)	0.56	0.47	0.45	0.60	0.44	0.47

We also analyze the subjective results according to melody awareness. The results for the two-way analysis of variance (ANOVA) show that melody awareness and method type significantly affect all metric scores ($p < 0.05$). All models achieve a higher P score than without awareness with melody awareness. In particular, VTHarm and rVTHarm show higher

[2] Every experimental protocol was approved by the Institutional Review Board (IRB) of Seoul National University. Written consent forms were collected from the participants, and the study was conducted according to the ethical standards outlined in the 1962 Helsinki Declaration.

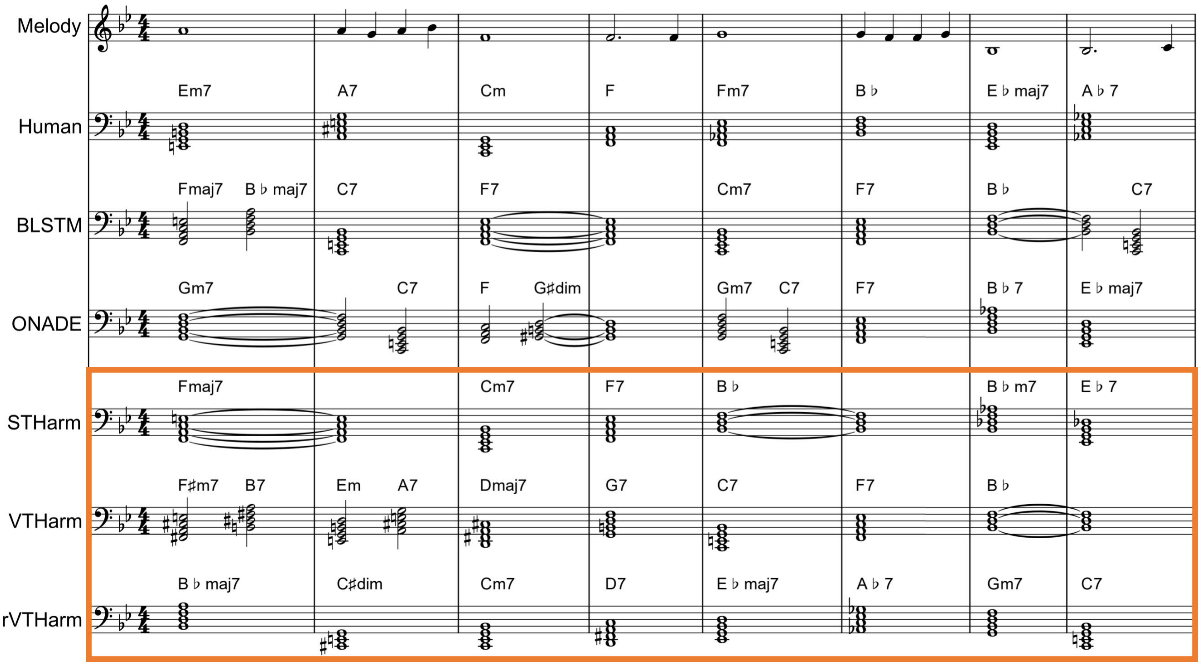


FIGURE 4. The generated samples of the five models and the human-composed chords given the melody from the song "Stella by Starlight." The orange box emphasizes the results from the three proposed models in which the harmonic rhythms follow the binary metrical structure. In contrast, the baseline models show the syncopated rhythms for some chords.

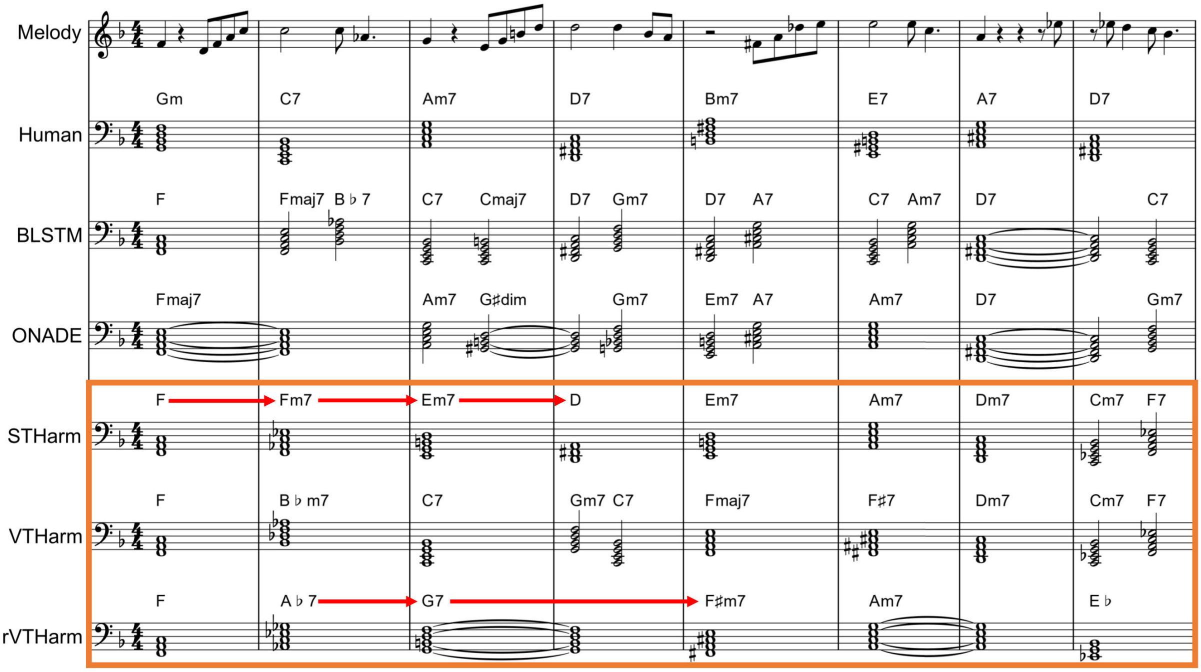


FIGURE 5. The generated samples of the five models and the human-composed chords given the melody from the song "Shiny Stockings". The orange box focuses on the results from the three proposed models in which the chord roots progress along the circle-of-fifths rule. The red arrows indicate the chromatic progressions where the chord notes descend or ascend by intervals of a major or minor second. These progressions are related to the given melody, where a certain pattern also develops chromatically.

P scores than ONADE, whereas they have similar or higher U and C scores to ONADE. This implies that the participants perceive the samples from the VT models to be more plausible than the baseline models when they know the melody, even though the VT models have comparable complexity and unexpectedness to the baseline models.

TABLE 7. Evaluation results of the chord similarity metrics according to adding the condition token *c*. VT and rVT denote VTHarm and rVTHarm, respectively.

Dataset	Chord Melody Dataset		
Metric	LD↓	TPSD↓	DICD↓
VT w/o c	0.90(±0.12)	2.76(±1.02)	122.75(±36.91)
VT w/ c	**0.86(±0.14)**	**2.72(±1.02)**	**121.08(±36.82)**
rVT w/o c	0.93(±0.10)	2.86(±1.32)	124.14(±35.79)
rVT w/ c	**0.86(±0.15)**	**2.71(±1.17)**	**118.01(±36.59)**
Dataset	Hooktheory Lead Sheet Dataset		
Metric	LD↓	TPSD↓	DICD↓
VT w/o c	0.79(±0.15)	**2.53(±1.13)**	100.02(±33.35)
VT w/ c	**0.77(±0.16)**	2.54(±1.15)	**98.55(±33.53)**
rVT w/o c	0.80(±0.16)	2.39(±1.21)	93.90(±34.65)
rVT w/ c	**0.79(±0.16)**	**2.32(±1.26)**	**91.63(±34.92)**

When the melody is unaware, BLSTM and rVTHarm obtain significantly lower Preference scores than when the melody is aware ($p < 0.001$). We further compute Pearson's correlation coefficient of U with C or P scores, as shown in Table 6. As a result, rVTHarm reveals the most negative correlation of U with both C and P scores when the melody is aware. This indicates that 1) controlled chords are more unexpected *and* unpleasant with a familiar melody, and 2) some factors other than complexity seem to cause an increased unexpectedness in rVTHarm. However, the mean preference score of rVTHarm significantly increases with melody awareness. This implies that the familiarity of the melody may strongly compensate for the high unexpectedness of rVTHarm. This tendency needs further investigation to improve the robustness of controllable melody harmonization.

E. QUALITATIVE RESULTS

Figs. 4 and 5 show some of the actual samples from the listening test for all five models as well as the human-composed music. These samples reveal the strengths of the proposed models. First, Fig. 4 mainly shows that the proposed models tend to reproduce the binary metrical structure of the chords compared to the baseline models. The binary metric structure is close to real-world music, most of which has been composed of four beats and strongly influenced by metrical boundaries [52]. In contrast, the chords generated from the baseline models show some syncopated rhythms, which can weaken the metrical boundaries. Fig. 5 illustrates another advantage of the proposed models, which is that the majority of the chord roots tend to shift in intervals either of perfect fourth or fifth according to the circle-of-fifths rule. This aspect reflects conventional Western music theory, which serves as domain knowledge for modeling real-world music [51], [54]. Moreover, the proposed models are shown to generate some natural chromatic progressions according to the given melody. On the other hand, the baseline models show some short transitions on the circle-of-fifths at arbitrary spots, in contrast to the melody with regular phrasings.

F. ABLATION STUDY

We conduct an ablation study to verify the benefit of adding the conditional token c to VTHarm and rVTHarm. We assume that c provides key signature information that can efficiently constrain the latent space to a concrete harmonic context, improving the chord structuredness and reconstruction performance of the model. We compute the chord similarity metrics between the ground truth and generated chords from the VT models according to the presence of c. The results are demonstrated in Table 7. This table shows that the VT models without c mostly obtain worse scores for all similarity metrics than the models with c. This indicates that adding key signature information to the VT models in most cases not only enhances the one-by-one accuracy but also improves the structure of the generated chords to be more human-like.

VI. CONCLUSION AND FUTURE WORK

In this paper, we have proposed melody harmonization models using the standard Transformer (STHarm), variational Transformer (VTHarm), and regularized variational Transformer (rVTHarm). We show that STHarm can create structured chords that are more human-like than LSTM-based models. VTHarm and rVTHarm can also generate more plausible chords than the baseline models with comparable chord diversity, especially when the melody is familiar. Furthermore, rVTHarm can control chord outputs with the disentangled representation for the intended attribute. Our study is limited to the shallow investigation of the connection between controllable attributes and melody awareness. Therefore, we plan to deeply explore the effect of melody awareness for more persuasive melody harmonization.

REFERENCES

[1] D. Makris, I. Karydis, and S. Sioutas, "Automatic melodic harmonization: An overview, challenges and future directions," in *Trends in Music Information Seeking, Behavior, and Retrieval for Creativity*. Hershey, PA, USA: IGI Global, 2016.

[2] S. A. Raczyński, S. Fukayama, and E. Vincent, "Melody harmonization with interpolated probabilistic models," *J. New Music Res.*, vol. 42, no. 3, pp. 223–235, Sep. 2013.

[3] I. Simon, D. Morris, and S. Basu, "MySong: Automatic accompaniment generation for vocal melodies," in *Proc. 27th Annu. CHI Conf. Hum. Factors Comput. Syst. (CHI)*, 2008, pp. 725–734.

[4] J.-F. Paiement, D. Eck, and S. Bengio, "Probabilistic melodic harmonization," in *Proc. 19th Conf. Can. Soc. Comput. Studies Intell.*, 2006, pp. 218–229.

[5] M. J. Steedman, "The blues and the abstract truth: Music and mental models," in *Mental Models in Cognitive Science*. NJ, USA: Lawrence Erlbaum Associates, 1996, pp. 305–318.

[6] A. R. R. Freitas and F. G. Guimaraes, "Melody harmonization in evolutionary music using multiobjective genetic algorithms," in *Proc. 8th Sound Music Comput. Conf. (SMC)*, Padova, Italy, 2011, pp. 1–8.

[7] M. Kaliakatsos-Papakostas and E. Cambouropoulos, "Probabilistic harmonization with fixed intermediate chord constraints," in *Proc. 40th ICMC*, Athens, Greece, 2014, pp. 1–8.

[8] H. Tsushima, E. Nakamura, K. Itoyama, and K. Yoshii, "Function- and rhythm-aware melody harmonization based on tree-structured parsing and split-merge sampling of chord sequences," in *Proc. 18th ISMIR*, Suzhou, China, 2017, pp. 1–7.

[9] H. Lim, S. Rhyu, and K. Lee, "Chord generation from symbolic melody using BLSTM networks," in *Proc. 18th ISMIR*, 2017, pp. 1–7.

[10] Y.-C. Yeh, W.-Y. Hsiao, S. Fukayama, T. Kitahara, B. Genchel, H.-M. Liu, H.-W. Dong, Y. Chen, T. Leong, and Y.-H. Yang, "Automatic melody harmonization with triad chords: A comparative study," *J. New Music Res.*, vol. 50, no. 1, pp. 37–51, Jan. 2021.

[11] C.-E. Sun, Y.-W. Chen, H.-S. Lee, Y.-H. Chen, and H.-M. Wang, "Melody harmonization using orderless NADE, chord balancing, and blocked Gibbs sampling," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 4145–4149.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. 31st Conf. NeurIPS*, 2017, pp. 1–11.

[13] C. Anderson, D. Carlton, R. Miyakawa, and D. Schwachhofer. (2021). *Hooktheory*. Accessed: Sep. 5, 2021. [Online]. Available: https://www.hooktheory.com

[14] S. Hiehn. (2019). *Chord Melody Dataset*. Accessed: Sep. 5, 2021. [Online]. Available: https://github.com/shiehn/chord-melody-dataset

[15] C.-Z. A. Huang, A. Vaswani, J. Uszkoreit, N. Shazeer, I. Simon, C. Hawthorne, A. M. Dai, M. D. Hoffman, M. Dinculescu, and D. Eck, "Music transformer," 2018, *arXiv:1809.04281*.

[16] H. C. Longuet-Higgins and M. J. Steedman, "On interpreting bach," *Mach. Intell.*, vol. 6, pp. 221–241, Jan. 1971.

[17] Y.-S. Huang and Y.-H. Yang, "Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions," in *Proc. 28th ACM Int. Conf. Multimedia*, Oct. 2020, pp. 1180–1188.

[18] K. Choi, J. Park, W. Heo, S. Jeon, and J. Park, "Chord conditioned melody generation with transformer based decoders," *IEEE Access*, vol. 9, pp. 42071–42080, 2021, doi: 10.1109/ACCESS.2021.3065831.

[19] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[20] M. Tokumaru, K. Yamashita, N. Muranaka, and S. Imanishi, "Membership functions in automatic harmonization system," in *Proc. 28th IEEE Int. Symp. Multiple- Valued Log.*, May 1998, pp. 350–355.

[21] A. Roberts, J. Engel, C. Raffel, C. Hawthorne, and D. Eck, "A hierarchical latent vector model for learning long-term structure in music," in *Proc. 35th ICML*, Stockholm, Sweden, 2018, pp. 4364–4373.

[22] Z. Lin, G. I. Winata, P. Xu, Z. Liu, and P. Fung, "Variational transformers for diverse response generation," 2020, *arXiv:2003.12738*.

[23] B. Zhang, D. Xiong, J. Su, H. Duan, and M. Zhang, "Variational neural machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 124–141.

[24] X. Sheng, L. Xu, J. Guo, J. Liu, R. Zhao, and Y. Xu, "IntroVNMT: An introspective model for variational neural machine translation," in *Proc. 34th AAAI*, New York, NY, USA, 2020, pp. 8830–8837.

[25] X. Liu, J. Zhao, S. Sun, H. Liu, and H. Yang, "Variational multimodal machine translation with underlying semantic alignment," *Inf. Fusion*, vol. 69, pp. 73–80, May 2021.

[26] K. Choi, C. Hawthorne, I. Simon, M. Dinculescu, and J. Engel, "Encoding musical style with transformer autoencoders," in *Proc. 37th ICML*, 2020, pp. 1899–1908.

[27] J. Jiang, G. G. Xia, D. B. Carlton, C. N. Anderson, and R. H. Miyakawa, "Transformer VAE: A hierarchical model for structure-aware and interpretable music representation learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 516–520.

[28] R. Yang, D. Wang, Z. Wang, T. Chen, J. Jiang, and G. Xia, "Deep music analogy via latent representation disentanglement," in *Proc. 20th ISMIR*, Delft, The Netherlands, 2019, pp. 596–603.

[29] T. Akama, "Controlling symbolic music generation based on concept learning from domain knowledge," in *Proc. 20th ISMIR*, Delft, The Netherlands, 2019, pp. 816–823.

[30] H. H. Tan and D. Herremans, "Music fadernets: Controllable music generation based on high-level features via low-level feature modelling," in *Proc. 21st ISMIR*, Montreal, QC, Canada, 2020, pp. 1–8.

[31] Z. Wang, Y. Zhang, Y. Zhang, J. Jiang, R. Yang, J. Zhao, and G. Xia, "Pianotree VAE: Structured representation learning for polyphonic music," in *Proc. 21st ISMIR*, Montreal, QC, Canada, 2020, pp. 1–8.

[32] A. Pati and A. Lerch, "Latent space regularization for explicit control of musical attributes," in *Proc. 36th ICML*, 2019, pp. 1–3.

[33] P. R. Illescas, D. Rizo, and J. M. Quereda, "Harmonic, melodic, and functional automatic analysis," in *Proc. 33rd ICMC*, Copenhagen, Denmark, 2007, pp. 1–7.

[34] H. V. Koops, J. P. Magalhães, and W. B. de Haas, "A functional approach to automatic melody harmonisation," in *Proc. 1st ACM SIGPLAN Workshop Funct. Art, Music, Modeling Design (FARM)*, 2013, pp. 47–58.

[35] H. Tsushima, E. Nakamura, and K. Yoshii, "Bayesian melody harmonization based on a tree-structured generative model of chord sequences and melodies," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, pp. 1644–1655, 2020.

[36] S. Oore, I. Simon, S. Dieleman, D. Eck, and K. Simonyan, "This time with feeling: Learning expressive musical performance," *Neural Comput. Appl.*, vol. 32, no. 4, pp. 955–967, Feb. 2020.

[37] C. Donahue, H. H. Mao, Y. E. Li, G. W. Cottrell, and J. McAuley, "LakhNES: Improving multi-instrumental music generation with cross-domain pre-training," in *Proc. 20th ISMIR*, Delft, The Netherlands, 2019, pp. 1–8.

[38] S.-L. Wu and Y.-H. Yang, "The Jazz transformer on the front line: Exploring the shortcomings of AI-composed music through quantitative measures," in *Proc. 21st ISMIR*, Montreal, QC, Canada, 2020, pp. 1–8.

[39] Z. Wang, D. Wang, Y. Zhang, and G. Xia, "Learning interpretable representation for controllable polyphonic music generation," in *Proc. 21st ISMIR*, Montreal, QC, Canada, 2020, pp. 1–8.

[40] N. Li, S. Liu, Y. Liu, S. Zhao, M. Liu, and M. Zhou, "Neural speech synthesis with transformer network," in *Proc. 33rd AAAI*, 2019, pp. 6706–6713.

[41] C. Raphael and J. Stoddard, "Functional harmonic analysis using probabilistic models," *Comput. Music J.*, vol. 28, no. 3, pp. 45–52, Sep. 2004.

[42] K. Sohn, X. Yan, and H. Lee, "Learning structured output representation using deep conditional generative models," in *Proc. 28th NeurIPS*, Montreal, QC, Canada, 2015, pp. 1–9.

[43] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner, "β-VAE: Learning basic visual concepts with a constrained variational framework," in *Proc. 5th ICLR*, Toulon, France, 2017, pp. 1–22.

[44] A. Roberts and C. F. Hawthorne. (2021). *Magenta Musicxml Parser*. Accessed: Sep. 5, 2021. [Online]. Available: https://github.com/magenta/note-seq/blob/main/note_seq/musicxml_parser.%py

[45] D. Jeong, T. Kwon, and J. Nam, "VirtuosoNet: A hierarchical attention RNN for generating expressive piano performance from music score," in *Proc. 32nd NeurIPS*, Montreal, QC, Canada 2018, pp. 1–5.

[46] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[47] G. Brunner, A. Konrad, Y. Wang, and R. Wattenhofer, "MIDI-VAE: Modeling dynamics and instrumentation of music with applications to style transfer," in *Proc. 21st ISMIR*, Paris, France, 2018, pp. 1–8.

[48] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. 17th Assoc. Comput. Linguist. (NAACL)*, 2018, pp. 1–6.

[49] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *Proc. 7th ICLR*, 2019, pp. 1–39.

[50] T. Fujishima, "Realtime chord recognition of musical sound: A system using common lisp music," in *Proc. ICMC*, Beijing, China, 1999, pp. 464–467.

[51] C. Harte, M. Sandler, and M. Gasser, "Detecting harmonic change in musical audio," in *Proc. 1st ACM Workshop Audio Music Comput. Multimedia (AMCMM)*, 2006, pp. 21–26.

[52] J.-F. Paiement, D. Eck, and S. Bengio, "A probabilistic model for chord progressions," in *Proc. 6th ISMIR*, London, U.K., 2005, pp. 1–14.

[53] W. B. de Haas, F. Wiering, and R. C. Veltkamp, "A geometrical distance measure for determining the similarity of musical harmony," *Int. J. Multimedia Inf. Retr.*, vol. 2, no. 3, pp. 189–202, Sep. 2013.

[54] F. Lerdahl, "Tonal pitch space," *Music Perception*, vol. 5, no. 3, pp. 315–349, Apr. 1988.

[55] E. Cambouropoulos, "A directional interval class representation of chord transitions," in *Proc. 12th Int. Conf. Music Percept. Cogn. (ICMPC)*, Thessaloniki, Greece, 2012, pp. 1–5.