AUDIO-TO-SYMBOLIC ARRANGEMENT VIA CROSS-MODAL MUSIC REPRESENTATION LEARNING

Ziyu Wang[12]   *Dejing Xu*[2]   *Gus Xia*[1]   *Ying Shan*[2]

[1]Music X Lab, NYU Shanghai   [2]ARC Lab, Tencent PCG

ABSTRACT

Could we automatically derive the score of a piano accompaniment based on the audio of a pop song? This is the audio-to-symbolic *arrangement* problem we tackle in this paper. A good arrangement model should not only consider the audio content but also have prior knowledge of piano composition (so that the generation "sounds like" the audio and meanwhile maintains musicality). To this end, we contribute a cross-modal representation-learning model, which 1) extracts chord and melodic information from the audio, and 2) learns texture representation from both audio and a *corrupted* ground truth arrangement. We further introduce a tailored training strategy that gradually shifts the source of texture information from corrupted score to audio. In the end, the score-based texture posterior is reduced to a standard normal distribution, and only audio is needed for inference. Experiments show that our model captures major audio information and outperforms baselines in generation quality. [1]

Index Terms— cross-modal representation, automatic arrangement, disentangled representation

1. INTRODUCTION

Piano arrangement is widely used in practice to reproduce various complex music signals. The key idea is to transform the original music, usually represented by an audio mixture or full score of a band, into a piano score (that can be performed using only 2 or 4 hands) without loss of major music information. For example, piano reductions are made for classical orchestral music and piano covers are created for pop songs. A good arrangement is not merely a transcription of the original audio mapped onto the keyboard, but also a realization of the original content that makes musical sense as a composition in the new instrument.

In the paper, our focus is automatic *audio-to-symbolic* arrangement. Unlike automatic music transcription [1, 2], the task aims at the symbolic generation according to the audio content which consists of an arbitrary set of instruments and may contain timbral effects that cannot be easily transcribed.

---

[1]Code and demos can be accessed via `https://github.com/ZZWaang/audio2midi`.

Existing systems mainly rely on rule-based or simple statistical models [3, 4, 5, 6], where audio-analysis and symbolic-generation modules are more or less independent and loosely connected by some bottleneck audio features. Such design often suffers from two limitations. First, the arrangement patterns are usually very rigid since both the bottleneck audio features and the accompaniment patterns are largely pre-defined. Second, nuanced features such as groove patterns and bass lines are difficult to be modeled using existing MIR techniques.

We consider end-to-end audio-to-symbolic deep generative modeling a better choice, as neural-based modeling potentially enables more flexible symbolic generation and an end-to-end architecture allows nuanced features to flow from audio to symbolic modal. On the other hand, we are faced with a great challenge — the relation between audio and its possible symbolic arrangement is essentially *one-to-many* and the prior knowledge of a good piano composition is only partially present in the audio. In other words, a naive supervised-learning model would easily get confused by the noisy audio-symbolic pairs and collapse to certain specific accompaniment patterns.

To solve the problem, we propose a cross-modal representation learning framework, in which the input is the audio of a pop-song accompaniment under arbitrary instrumentation, while the output is an arrangement in MIDI format. The model encodes a cross-modal representation from both audio and symbolic modals and decodes the information back to the symbolic domain. The latent representation contains the audio content and reflects prior knowledge about symbolic composition. During pre-training, we initialize the model to be an almost pure symbolic-to-symbolic variational autoencoder. By gradually corrupting the input and strengthening the variational constraints, the model is trained to lean more towards the audio. During the fine-tuning, we could provide the symbolic side with either Gaussian noise or information from previous bars to make the model fully dependent on audio or being autoregressive.

In sum, we contribute the first end-to-end approach for automatic audio-to-symbolic arrangement. The quality of the generated samples is significantly higher than baselines and even rated higher than human compositions in terms of creativity. Moreover, the arrangement problem is tackled by a

(a) Model architecture.
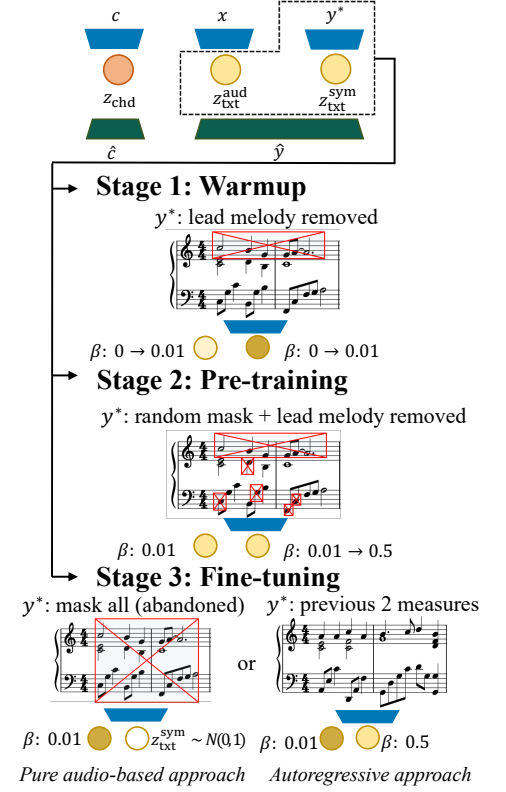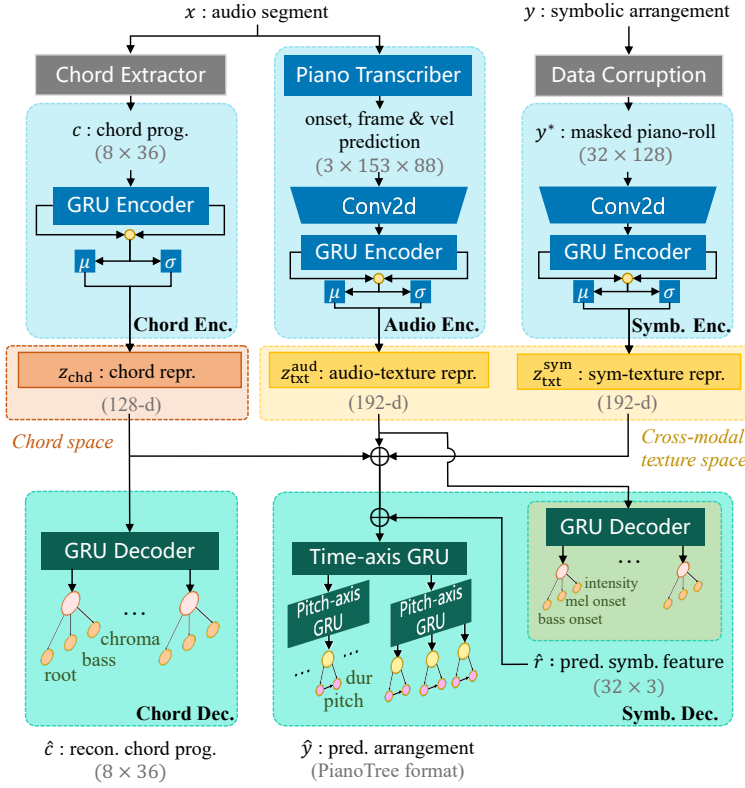
(b) Training strategy.

Fig. 1. The proposed model architecture and training strategy.

tailored training strategy that optimizes the supervised objective under an unsupervised cross-modal representation learning framework, which can be potentially generalized to other one-to-many supervised training tasks.

2. RELATED WORK

Automatic arrangement tasks can be conducted based on either symbolic or audio sources. Pure symbolic arrangement is commonly studied using deep generative models and has achieved considerable progress [7, 8, 9, 10]. In contrast, *audio-to-symbolic* arrangement, which is more related to this paper, is still underresearched. Existing methods are mostly rule-based or rely on hand-crafted statistics. E.g., Takamori et al. extract chords and melodies from audio and pre-define several accompaniment textures [4, 5]. Song2Quartet [3] and Song2Guitar[6] further introduce matching probabilities between audio and score and use dynamic programming to search the notes. These models often lead to rigid patterns and the musicality cannot yet serve for practical purposes.

An alternative shortcut to achieve arrangement is timbre style transfer [11, 12, 13, 14, 15], in which a latent timbre space is first learned and then transferred to piano timbre

during inference. However, existing models are either constrained to monophonic instruments or based on synthetic audio data. Real-world audio is more complicated in instrumentation and the integration of audio content with piano composition techniques is still an open problem.

3. METHOD

We aim to learn *chord representation*, *audio-texture representation* from audio and *symbolic-texture representation* from its paired symbolic arrangement. In this paper, we consider 2-bar audio segments (provided with beat annotation) and symbolic arrangement in $\frac{4}{4}$ time signature. The symbolic arrangement is represented under $\frac{1}{4}$ beat resolution.

3.1. Model Architecture

Figure 1(a) shows the overall model architecture which adopts an encoder-decoder architecture and contains five parts: 1) a chord encoder, 2) a chord decoder, 3) an audio encoder, 4) a symbolic encoder, and 5) a symbolic decoder. We consider our model a cross-modal extension of a symbolic-domain chord and texture disentanglement study [7].

The chord encoder adopts a GRU layer to encode 128-d chord representation z_{chd} from a chord progression, which is extracted from the audio using an existing chord extraction algorithm [16]. The chord decoder which reconstructs the input chord progression is introduced as a mechanism to avoid *posterior collapse* of z_{chd}.

The input to the audio encoder is an 8-beat long audio segment, time-stretched and resampled to 95 BPM with a sample rate of 16kHz. We first use a piano transcriber architecture [17] to embed the audio feature into a stack of piano-roll-like matrices of onset, frame and velocity predictions. Then, we use a 2D convolution layer followed by a GRU layer to encode 192-d audio-texture representation $z_{\text{txt}}^{\text{aud}}$. The same model structure is applied to the symbolic encoder to extract 192-d symbolic-texture representation $z_{\text{txt}}^{\text{sym}}$ from a corrupted ground truth piano-roll. The data corruption is controlled by a tailored training strategy introduced in section 3.3.

The symbolic decoder takes in the concatenation of z_{chd}, $z_{\text{txt}}^{\text{aud}}$ and $z_{\text{txt}}^{\text{sym}}$ and decodes the symbolic arrangement in a hierarchical manner using the decoder module of PianoTree VAE [18], the state-of-the-art polyphonic representation learning model. Besides the latent codes, the PianoTree decoder also takes in a time series of symbolic features, which is predicted from $z_{\text{txt}}^{\text{aud}}$ *only* to enhance audio information retrieval. Specifically, we explicitly constrain $z_{\text{txt}}^{\text{aud}}$ to predict three symbolic features: *bass onset*, *melody onset*, and *rhythmic intensity*, which usually strongly correlate with audio rhythmic information of bass drum, lead melody and groove patterns, respectively. Both bass onset and melody onset are time series of onset probabilities, and rhythmic intensity is a time series of scalar values. The predicted features are fed to the corresponding time steps of the time-axis GRU in the PianoTree decoder. Similar method is also used to achieve disentanglement in [19].

3.2. Training Objective

The loss terms in our model include 1) reconstruction losses of chord, arrangement, and symbolic features, and 2) KL losses between all three latent factors with standard normal distributions. Our model is essentially a *conditional variational autoencder*, since the loss function can be formalized as the *evidence lower bound* (ELBO) of the conditional probability $p(y|x)$, where x is the audio and y is the arrangement.

The posterior distribution of the conditional VAE is defined as the product of the three encoder models:

$$q_{\boldsymbol{\phi}}(\mathbf{z}|x,y) := q_{\phi_1}(z_{\text{chd}}|c)q_{\phi_2}(z_{\text{txt}}^{\text{aud}}|x)q_{\phi_3}(z_{\text{txt}}^{\text{sym}}|y),$$

where $\mathbf{z} := [z_{\text{chd}}, z_{\text{txt}}^{\text{aud}}, z_{\text{txt}}^{\text{sym}}]$, and $\boldsymbol{\phi} := [\phi_1, \phi_2, \phi_3]$ denotes the encoder parameters. Note that the chord progression c is a deterministic transform from x and is therefore absent in $q_{\boldsymbol{\phi}}(\mathbf{z}|x,y)$. The reconstruction distribution is defined as the product of the three reconstruction terms:

$$p_{\boldsymbol{\theta}}(x|\mathbf{z}) := p_{\theta_1}(c|z_{\text{chd}})p_{\theta_2}(y|\mathbf{z},r)p_{\theta_3}(r|z_{\text{txt}}^{\text{aud}}),$$

where $\boldsymbol{\theta} := [\theta_1, \theta_2, \theta_3]$ denotes the decoder parameters, r denotes the ground truth symbolic features, and $p_{\theta_1}(c|z_{\text{chd}})$ is interpreted as a regularizer to the output distribution. Finally, the loss function is:

$$\mathcal{L}_{\beta}(\boldsymbol{\theta}, \boldsymbol{\phi}; x) = -\mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\phi}}(\mathbf{z}|x,y)} \Big[\log p_{\boldsymbol{\theta}}(x|\mathbf{z}) \Big]$$
$$+ \beta \, \text{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|x,y) || p(\mathbf{z})),$$

where $p(\mathbf{z})$ is a 512-d standard normal prior and β is the KL annealing parameter.

3.3. Training Strategy

We propose a training strategy (shown in Figure 1(b)) to balance information from the audio encoder and the symbolic encoder so that the training starts with the unsupervised symbolic reconstruction task and shifts to the supervised audio-to-symbolic task. Particularly, there are three stages:

Stage 1, Warm-up: The lead voice of the ground truth arrangement is masked, and β increases from 0 to 0.01 for all three latent factors. The model is therefore enforced to predict melody solely from the audio.

Stage 2, Pre-training: Besides melody, the rest of the notes are randomly masked under probability ranging from 0.5-0.8, where lower pitches have a higher probability to be masked. Meanwhile, β increases from 0.01 to 0.5 for $z_{\text{txt}}^{\text{sym}}$ and keeps 0.01 for the other two factors. The model is expected to learn more information from the audio.

Stage 3, Fine-tuning: The model can be purely audio-dependent at this stage: we completely abandon the symbolic encoder by sampling $z_{\text{txt}}^{\text{sym}}$ from a standard normal distribution. Alternatively, we can also feed the arrangement of the previous two measures to the symbolic encoder to make the model autoregressive.

4. EXPERIMENTS

4.1. Implementation Detail

We train our model on the POP909 dataset [20], which contains about 1K MIDI files of pop song arrangements with time-aligned audios. We use the piano accompaniment MIDI tracks and keep the pieces with $\frac{2}{4}$ and $\frac{4}{4}$ meters and cut them into 8-beat music segments. The audio is also sliced into 8-beat segments and the vocal is removed by the Spleeter source separation algorithm [21]. In all, we have 66K samples. We randomly split the dataset (at song level) into training set (90%) and test set (10%). All training samples are further augmented by transposing to all 12 keys. The chord, beat, and melody track annotations are all included in the dataset while the ground truth bass onset is defined to be the occurrence of MIDI pitch lower than 48, and rhythmic intensity is the number of simultaneous onsets normalized by a constant.
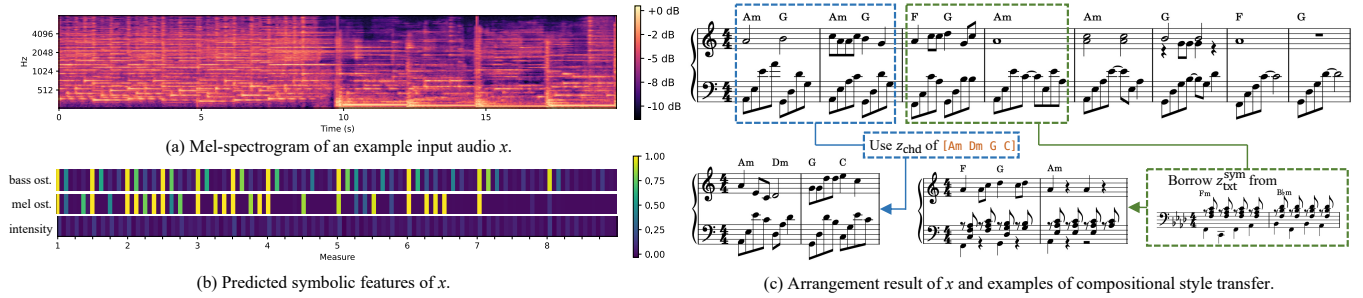
(a) Mel-spectrogram of an example input audio x.

(b) Predicted symbolic features of x.

(c) Arrangement result of x and examples of compositional style transfer.

Fig. 2. An example of automatic arrangement based on the audio of an 8-bar excerpt from *1001 Nights* by Samuel Tai.

The piano transcriber used in our model is pre-trained on the MAESTRO dataset [22]. The model contains 62M trainable parameters in total, including 26M parameters in the piano transcriber. We use a batch size of 64 and Adam optimizer [23] with a scheduled learning rate from 4e-4 to 6e-4.

4.2. Arrangement Example

A 16-bar arrangement example (by predicting every 2 bars independently) is shown in Figure 2. The audio has a lead instrument and frequent bass note changes at the beginning, and a less intense groove halfway to the end (Figure 2(a)). These features are captured in the symbolic feature prediction (Figure 2(b)), as well as the symbolic arrangement (Figure 2(c)), where we see melody with arpeggio texture in mm. 1-4, and arpeggio with decreasing intensity in mm. 5-8.

We also demonstrate the model (after the pre-training stage) is capable of the *compositional style transfer* tasks [15] via replacement of the disentangled factors [7]. First, we change the chords in mm. 1-2 to another chord progression [Am, Dm, G, C] represented by z_{chd} (indicated by the blue arrow), and the generation changes to the desired progression while maintaining the original texture. Then, we replace mm. 3-4 with a new symbolic texture represented by z_{txt}^{sym} (indicated by the green arrow), and the left-hand texture changes correspondingly while the harmony and the right-hand melody contour are kept unchanged.

4.3. Subjective Evaluation

We compare our proposed method with three baselines. The first two baselines adopt the common supervised approach, implemented with only the audio encoder and the PianoTree decoder with or without KL loss. The third baseline is solely chord-dependent, by setting z_{txt}^{aud} to zero.

We invite people to subjectively rate the generation quality through a double-blind online survey. During the survey, the subjects listen to 6 groups of samples. In each group, the original audio is played, followed by the generated samples and the ground truth composition in random order. Both the order of groups and the sample order within each group are
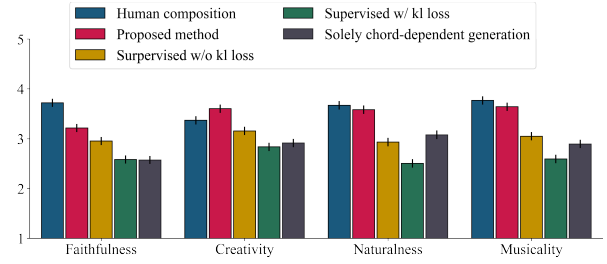

Fig. 3. Subjective evaluation results.

randomized. After listening to each sample, the subjects rate them based on a 5-point scale from 1 (very low) to 5 (very high) according to four criteria: *faithfulness* (to the original audio), *creativity*, *naturalness* and overall *musicality*.

A total of 26 subjects (8 females and 18 males) with different musical backgrounds have completed the survey. Figure 3 shows the result where the heights of the bars represent the means of the ratings and the error bars represent the confidence interval computed via within-subject ANOVA. The result shows that the proposed model is significantly better than the baseline models in terms of all four criteria, and the creativity is even significantly better than human composition (with p-value < 0.005).

5. CONCLUSION

We have contributed a cross-modal representation learning framework as the first end-to-end approach to accomplish the audio-to-symbolic automatic arrangement problem. Experimental results show that our model is able to capture harmonies, melody lines, and groove patterns from the audio without loss of musicality. The main novelty lies in the cross-modal training strategy that gradually shifts the input source from one modal to the other. We see such kind of tailored self-supervision control as a bridge between unsupervised learning tasks and supervised training. In the future, we seek more flexible cross-modal methods and make the audio-to-symbolic conversion more controllable.

6. REFERENCES

[1] Emmanouil Benetos, Simon Dixon, Dimitrios Gian-noulis, Holger Kirchhoff, and Anssi Klapuri, "Automatic music transcription: challenges and future directions," *JIIS*, vol. 41, no. 3, pp. 407–434, 2013.

[2] Emmanouil Benetos, Simon Dixon, Zhiyao Duan, and Sebastian Ewert, "Automatic music transcription: An overview," *IEEE Signal Processing Magazine*, vol. 36, no. 1, pp. 20–30, 2018.

[3] Graham Percival, Satoru Fukayama, and Masataka Goto, "Song2quartet: A system for generating string quartet cover songs from polyphonic audio of popular music.," in *ISMIR*, 2015, pp. 114–120.

[4] Hirofumi Takamori, Haruki Sato, Takayuki Nakatsuka, and Shigeo Morishima, "Automatic arranging musical score for piano using important musical elements," in *SMC*, 2017, pp. 35–41.

[5] Hirofumi Takamori, Takayuki Nakatsuka, Satoru Fukayama, Masataka Goto, and Shigeo Morishima, "Audio-based automatic generation of a piano reduction score by considering the musical structure," in *MMM*, 2019, pp. 169–181.

[6] Shunya Ariga, Satoru Fukayama, and Masataka Goto, "Song2guitar: A difficulty-aware arrangement system for generating guitar solo covers from polyphonic audio of popular music.," in *ISMIR*, 2017, pp. 568–574.

[7] Ziyu Wang, Dingsu Wang, Yixiao Zhang, and Gus Xia, "Learning interpretable representation for controllable polyphonic music generation," in *ISMIR*, 2020, pp. 662–669.

[8] Ian Simon, Adam Roberts, Colin Raffel, Jesse Engel, Curtis Hawthorne, and Douglas Eck, "Learning a latent space of multitrack measures," *arXiv preprint arXiv:1806.00195*, 2018.

[9] Jean-Pierre Briot, Gaëtan Hadjeres, and François-David Pachet, "Deep learning techniques for music generation–a survey," *arXiv preprint arXiv:1709.01620*, 2017.

[10] Ke Chen, Cheng-i Wang, Taylor Berg-Kirkpatrick, and Shlomo Dubnov, "Music sketchnet: Controllable music generation via factorized representations of pitch and rhythm," in *ISMIR*, 2020, pp. 77–84.

[11] Jesse Engel, Lamtharn (Hanoi) Hantrakul, Chenjie Gu, and Adam Roberts, "Ddsp: Differentiable digital signal processing," in *ICLR*, 2020.

[12] Bryan Wang and Yi-Hsuan Yang, "Performancenet: Score-to-audio music generation with multi-band convolutional residual network," in *AAAI*, 2019, vol. 33, pp. 1174–1181.

[13] Yun-Ning Hung, I-Tung Chiang, Yi-An Chen, and Yi-Hsuan Yang, "Musical composition style transfer via disentangled timbre representations," in *IJCAI*, 2019, pp. 4697–4703.

[14] Liwei Lin, Gus Xia, Qiuqiang Kong, and Junyan Jiang, "A unified model for zero-shot music source separation, transcription and synthesis," in *ISMIR*, 2021, pp. 381–388.

[15] Gus G. Xia and Shuqi Dai, "Music style transfer: A position paper," in *MUME*, 2018, p. 6.

[16] Junyan Jiang, Ke Chen, Wei Li, and Gus Xia, "Large-vocabulary chord transcription via chord structure decomposition," in *ISMIR*, 2019, pp. 644–651.

[17] Curtis Hawthorne, Erich Elsen, Jialin Song, Adam Roberts, Ian Simon, Colin Raffel, Jesse H. Engel, Sageev Oore, and Douglas Eck, "Onsets and frames: Dual-objective piano transcription," in *ISMIR*, 2018, pp. 50–57.

[18] Ziyu Wang, Yiyi Zhang, Yixiao Zhang, Junyan Jiang, Ruihan Yang, Gus Xia, and Junbo Zhao, "PianoTree VAE: structured representation learning for polyphonic music," in *ISMIR*, 2020, pp. 368–375.

[19] Ruihan Yang, Dingsu Wang, Ziyu Wang, Tianyao Chen, Junyan Jiang, and Gus Xia, "Deep music analogy via latent representation disentanglement," in *ISMIR*, 2019, pp. 596–603.

[20] Ziyu Wang, Ke Chen, Junyan Jiang, Yiyi Zhang, Maoran Xu, Shuqi Dai, and Gus Xia, "POP909: A pop-song dataset for music arrangement generation," in *ISMIR*, 2020, pp. 38–45.

[21] Romain Hennequin, Anis Khlif, Felix Voituret, and Manuel Moussallam, "Spleeter: a fast and efficient music source separation tool with pre-trained models," *JOSS*, vol. 5, no. 50, pp. 2154, 2020.

[22] Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, Ian Simon, Cheng-Zhi Anna Huang, Sander Dieleman, Erich Elsen, Jesse Engel, and Douglas Eck, "Enabling factorized piano music modeling and generation with the MAESTRO dataset," in *ICLR*, 2019.

[23] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.