Generating Structured Music through Self-Attention

Anna Huang [1 2] **Ashish Vaswani** [2] **Jakob Uszkoreit** [2] **Noam Shazeer** [2] **Andrew Dai** [2] **Matt Hoffman** [2] **Curtis Hawthorne** [2] **Douglas Eck** [2]

Abstract

Music relies heavily on self-reference to build structure and meaning. We explore the TRANSFORMER architecture (Vaswani et al., 2017) as a generative model for music, as self-attention has shown compelling results on tasks that require long-term structure such as Wikipedia summary generation (Liu et al., 2018). However, *timing* information is critical for polyphonic music, and TRANSFORMER does not explicitly represent absolute or relative timing in its structure. To address this challenge, Shaw et al. (2018) introduced relative position representations to self-attention to improve machine translation. However, the formulation was not scalable to longer sequences. We propose an improved formulation which reduces its memory requirements from $O(l^2d)$ to $O(ld)$, making it possible to train much longer sequences and achieve faster convergence [1]. In experiments with symbolic music generation, we find that relative self-attention substantially improves sample quality. When primed, the model generates continuations that develop the prime in a coherent fashion and exhibit long-term structure [2].

1. Introduction

A musical piece often consists of recurring themes, either explicitly laid out as e.g. verse-chorus-verse pairs or, more generally, as repetitions of musically-similar motives (i.e. short phrases). To generate a coherent piece, a model needs to remember the motives that came before, in order to repeat, vary and further develop them, and also to know how to create contrast and surprise. Self-attention mechanisms are a natural fit for this challenge, as they offer direct access to the generated history, allowing the model to choose the level of detail or summarization to derive from it.

We explore the self-attention based TRANSFORMER architecture (Vaswani et al., 2017) as a generative model for music. Timing information is critical for music, yet TRANSFORMER does not encode sequential ordering in its structure, i.e. each time step can freely attend to any past time step as if it were one time step away. The only source of timing information comes from the positional sinusoids added to the input embedding, which can be difficult to disentangle. Shaw et al. (2018) proposes to add a relative positional representation that allow for similarity comparisons to be sensitive to how far the tokens are apart in a sequence. However their implementation was not scalable to longer sequences.

This paper makes two main contributions. First, we improve the memory consumption of the relative attention mechanism proposed in Shaw et al. (2018), from $O(l^2d)$ to $O(ld)$, where l is the length of the sequence, and d the hidden size of the model. Second, we show in a series of experiments on music generation that relative attention is critical for various music representations as they all carry strong sequential dependencies. J.S. Bach chorales [3], a canonical dataset used for evaluating generative models for music (e.g. used in Allan & Williams (2005); Boulanger-Lewandowski et al. (2012); Liang (2016); Hadjeres et al. (2016), uses a serialized instrument/time grid-like representation, while an event-based representation is used for the Piano E-competition dataset [4] as in Simon & Oore (2017). In both cases, relative self-attention results in more consistency in sample quality for unconditioned generation and can generate sequences longer than those used in training. Given an initial motif, the model generates continuations that develop the motif in a coherent way over phrases. This illustrates the potential of relative self-attention becoming a productive tool for musicians.

[1] Google AI Resident [2] Google Brain. Correspondence to: Anna Huang <annahuang@google.com>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

[1] Code for improved relative attention: *dot_product_self_attention_relative_v2* at `https://github.com/tensorflow/tensor2tensor/blob/master/tensor2tensor/layers/common_attention.py`.

[2] Samples: `http://bit.ly/2LbdENq`

[3] J.S. Bach chorales dataset: `https://github.com/czhuang/JSB-Chorales-dataset`

[4] Piano E-competition dataset (under competition history): `http://www.piano-e-competition.com/`

2. Model

2.1. Background: TRANSFORMER

TRANSFORMER was originally formulated as a sequence-to-sequence model, using primarily attention mechanisms. The decoder by itself can be used as an unconditioned autoregressive generative model, by adopting self-attention that only attends to the past and not the future. More specifically, each position can attend to any subset of past positions, weighted by their relevance. To determine the weight of how much a previous time step $t-i$ informs the next-step prediction of the current time step t, a dot product is performed between the current query q_t and the previous key k_{t-i}. This is computed for every current and past position pairs and subsequently passed through a softmax. The normalized weights then determines how much the value v_{t-i} informs the hidden state z_t (i.e. Equation 1), which is then fed to a pointwise fully connected layer. For more details, see Vaswani et al. (2017).

$$Attention(Q,K,V) = softmax(QK^T)V \quad (1)$$

2.2. Improving Relative Self-Attention

Shaw et al. (2018) introduced relative positional representations to allow attention to be informed by how far two positions are apart in a sequence. This involves learning a relative positional embedding E_r, where each distance is given an embedding. This embedding is then used to modulate the relevance computation as shown in Equation 2.

$$Relative(Q,K,V,R) = softmax(QK^T + QR^T)V \quad (2)$$

The initial proposed implementation involves constructing an intermediate tensor R of shape (l,l,d) by first computing pairwise distances for each of the l by l positions and then gathering the corresponding embeddings from E_r. We observe that all of terms we need from QR^T are already available if we directly multiply Q and E_r. As Q is indexed by absolute query positions and E_r is indexed by relative distances, the result is an absolute-by-relative indexed tensor. To obtain an absolute-by-absolute indexed matrix that can be added to QK^T, we can skew QE_r by prepending an extra dummy column, reshaping the result to have an extra row and then deleting the first row (i.e. Figure 1). This reduces the memory requirements from $O(l^2d)$ to $O(ld)$ allowing us to train models with larger hidden sizes.

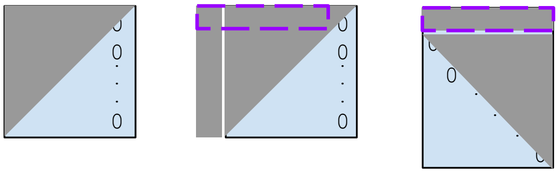

Figure 1. Steps for remapping absolute-by-relative indexed matrix to absolute-by-absolute indexed. The grey portions are not used.

3. Experiments

3.1. J.S. Bach Chorales

Four-part chorales are first discretized to a 16th-note grid, and then serialized by iterating through all the voices within a time step and then advancing time. As there is a direct correspondence between position in sequence and position on the timing/instrument grid in a piece, adding relative positional representations makes it easier to learn this grammar. This drastically improves negative loglikelihood (NLL) over baseline TRANSFORMER (Table 1), and gives rise to samples with more regular phrasings (bottom Figure 2).

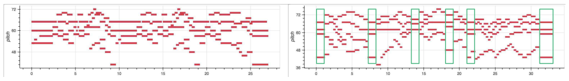

Figure 2. Comparing unconditioned generation from baseline self-attention (top) and relative self-attention (bottom).

Table 1. Validation NLL of baseline and relative self-attention.

Dataset	Baseline	Relative
JSB chorales	0.417	0.357
Piano E-competition	1.889	1.866

3.2. Piano E-competition

This dataset consists of performed classical piano music with expressive dynamics and timing, and is encoded using MIDI-like event-based representation (Simon & Oore, 2017). We compare the models by how they generate continuations to a given motif, with results shown in Figure 3. Relative attention reuses the given motif in a diverse set of ways, while baseline TRANSFORMER generates samples that are more uniform but all reusing the motif. LSTMs uses the motive initially but soon drifts off to other material.

Note the samples are generated at twice the length the models were trained on. Relative attention was able to generalize to lengths longer than trained but baseline TRANSFORMER deteriorates beyond its training length.

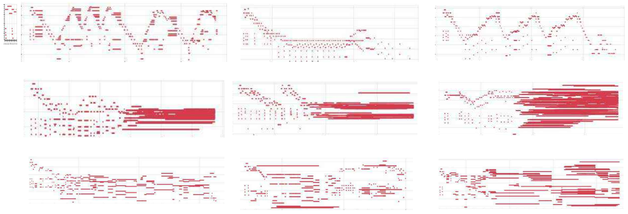

Figure 3. Continuations to a prime (top-left tiny score) are generated by TRANSFORMER with relative attention (top) where the samples repeats and varies the motif more, while less so in samples from baseline Transformer (middle) and LSTM (bottom).

Acknowledgements

We thank many colleagues for their help and insightful discussions: Tim Cooijmans, Lukasz Kaiser, Ryan Sepassi, Niki Parmar, Llion Jones, Vida Vakilotojar, Ian Simons, Jesse Engel, Adam Roberts, Erich Elsen, and Sander Dieleman.

References

Allan, M. and Williams, C. K. Harmonising chorales by probabilistic inference. *Advances in neural information processing systems*, 17:25–32, 2005.

Boulanger-Lewandowski, N., Bengio, Y., and Vincent, P. Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. *International Conference on Machine Learning*, 2012.

Hadjeres, G., Sakellariou, J., and Pachet, F. Style imitation and chord invention in polyphonic music with exponential families. *arXiv preprint arXiv:1609.05152*, 2016.

Liang, F. Bachbot: Automatic composition in the style of bach chorales. *Masters thesis, University of Cambridge*, 2016.

Liu, P. J., Saleh, M., Pot, E., Goodrich, B., Sepassi, R., Kaiser, L., and Shazeer, N. Generating wikipedia by summarizing long sequences. *arXiv preprint arXiv:1801.10198*, 2018.

Shaw, P., Uszkoreit, J., and Vaswani, A. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*, 2018.

Simon, I. and Oore, S. Performance rnn: Generating music with expressive timing and dynamics. https://magenta.tensorflow.org/performance-rnn, 2017.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, pp. 6000–6010, 2017.