COUNTERPOINT BY CONVOLUTION

Cheng-Zhi Anna Huang*
MILA, Université de Montréal
chengzhiannahuang@gmail.com

Tim Cooijmans†
MILA, Université de Montréal
tim.cooijmans@umontreal.ca

Adam Roberts
Google Brain
adarob@google.com

Aaron Courville
MILA, Université de Montréal
aaron.courville@umontreal.ca

Douglas Eck
Google Brain
deck@google.com

ABSTRACT

Machine learning models of music typically break down the task of composition into a chronological process, composing a piece of music in a single pass from beginning to end. On the contrary, human composers write music in a nonlinear fashion, scribbling motifs here and there, often revisiting choices previously made. We explore the use of blocked Gibbs sampling as an analogue to the human approach, and introduce COCONET, a convolutional neural network in the NADE family of generative models (Uria et al., 2016). Despite ostensibly sampling from the same distribution as the NADE ancestral sampling procedure, we find that a blocked Gibbs approach significantly improves sample quality. We provide evidence that this is due to some conditional distributions being poorly modeled. Moreover, we show that even the cheap approximate blocked Gibbs procedure from Yao et al. (2014) yields better samples than ancestral sampling. We demonstrate the versatility of our method on unconditioned polyphonic music generation.

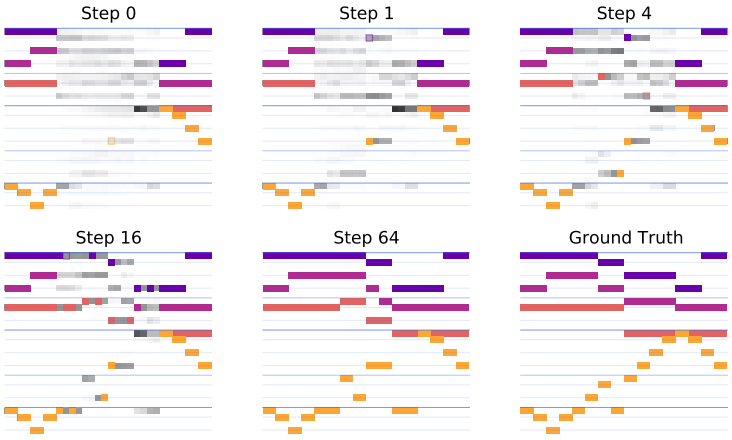

Figure 1: Ancestral inpainting of a corrupted Bach chorale by COCONET. Colors are used to distinguish the four voices. Grayscale heatmaps show predictions $p(\mathbf{x}_i \mid \mathbf{x}_C)$. The pitch sampled in the current step is indicated by a rectangular outline. The original Bach chorale is shown in the bottom right. Step 0 shows the corrupted Bach chorale. Step 64 shows the result.

*Work done while at Google Brain.
†Work done while at Google Brain.

1 INTRODUCTION

Machine learning can be used to create compelling art. This was shown recently by DeepDream (Mordvintsev et al., 2015), an optimization process that created psychedelic transformations of images. A similar idea underlies a variety of style transfer algorithms (Gatys et al., 2015), which impose textures and colors from one image onto another. More recently, the multistyle pastiche generator (Dumoulin et al., 2016) exposes adjustable knobs that allow users of the system fine-grained control over style transfers. Neural doodle (Champandard, 2016) further closes the feedback loop between algorithm and artist.

We wish to bring similar artistic tools to the domain of music. Whereas previous work in music has relied mainly on sequence models such as Hidden Markov Models (HMMs, Baum & Petrie (1966)) and Recurrent Neural Networks (RNNs, Rumelhart et al. (1988)), we instead employ convolutional neural networks due to their emphasis on capturing local structure and their invariance properties. Moreover, convolutional neural networks have shown to be extremely versatile once trained, as shown by a variety of creative uses in the literature (Mordvintsev et al., 2015; Gatys et al., 2015; Almahairi et al., 2015; Lamb et al., 2016).

We introduce COCONET, a deep convolutional model trained to reconstruct partial scores. Once trained, COCONET provides direct access to all conditionals of the form $p(\mathbf{x}_i \mid \mathbf{x}_C)$ where $\mathbf{x}_C$ is a fragment of a musical score $\mathbf{x}$ and $i \notin C$ is in its complement. Figure 1 shows an example of such conditionals used in completing a partial score.

COCONET is an instance of deep orderless NADE Uria et al. (2014), and thus learns an ensemble of factorizations of the joint $p(\mathbf{x})$. However, the sampling procedure for orderless NADE is not orderless. Sampling from an orderless NADE involves (randomly) choosing an ordering, and sampling ancestrally according to the chosen ordering. We have found that this produces poor results for the highly structured and complex domain of musical counterpoint.

Instead, we propose to use blocked-Gibbs sampling, essentially improving sample quality through rewriting. An instance of this was previously explored by Yao et al. (2014) who employed a NADE in the transition operator for a Markov Chain, yielding a Generative Stochastic Network (GSN). The transition consists of a corruption process that masks out a subset $\mathbf{x}_{\neg C}$ of variables, followed by a process that independently resamples variables $\mathbf{x}_i, i \notin C$ according to the distribution $p_\theta(\mathbf{x}_i \mid \mathbf{x}_C)$ emitted by the NADE. Crucially, the effects of independent sampling are amortized by annealing the probability with which variables are masked out. Whereas Yao et al. (2014) treat their procedure as a cheap approximation to ancestral sampling, we find that it produces superior samples.

We show the versatility of our method on unconditioned polyphonic music generation.

Section 2 discusses previous work in the area of automatic musical composition. The details of our model and training procedure are laid out in Section 3. In Section 4 we show that our approach is equivalent to that of deep and orderless NADE Uria et al. (2014). We discuss sampling from our model in Section 5. Results of quantitative and qualitative evaluations are reported in Section 6. Finally, Section 7 concludes.

2 RELATED WORK

Sequence models such as HMMs and RNNs are a natural choice for modeling music. However, one of the challenges in adapting such models to music is that music generally consists of multiple interdependent streams of events. This can be most clearly seen in the notion of counterpoint, which refers to the relationships between the movement of individual instruments in a musical work. Compare this to typical sequence domains such as speech and language, which involve modeling a single stream of events: a single speaker or a single stream of words.

Successful application of sequence models to music hence requires serializing or otherwise re-representing the music to fit the sequence paradigm. For instance, Liang (2016) serialize four-part Bach chorales by interleaving the parts, while Allan & Williams (2005) construct a chord vocabulary. Boulanger-Lewandowski et al. (2012) adopt a piano roll representation, which is a binary matrix $\mathbf{x}$ such that $\mathbf{x}_{it}$ is hot if some instrument is playing pitch i at time t. To model the joint probability distribution of the multi-hot pitch vector $\mathbf{x}_t$, they employ a Restricted Boltzmann Ma-

chine (RBM (Smolensky, 1986; Hinton et al., 2006)) or Neural Autoregressive Distribution Estimator (Uria et al., 2016) at each time step.

Moreover, the behavior of human composers does not fit the chronological mold assumed by previous authors. A human composer might start his work with a coarse chord progression and iteratively refine it, revisiting choices previously made. Sampling according to $x_t \sim p(x_t|x_{<t})$, as is common, cannot account for the kinds of timeless dependencies that composers employ. Hadjeres et al. (2016) sidestep the choice of causal factorization and instead employ an undirected Markov model to learn pairwise relationships between neighboring notes up to a specified number of steps away in a score. Sampling involves Markov Chain Monte Carlo (MCMC) using the model as a Metropolis-Hastings (MH) objective. The model permits constraints on the state space to support tasks such as melody harmonization. However, the Markov assumption severely limits the expressivity of the model.

We opt instead for a convolutional approach that avoids many of these issues and naturally captures both relationships across time and interactions between instruments.

3 COUNTERPOINT BY CONVOLUTION

We approach the task of music composition with a deep convolutional neural network (Krizhevsky et al., 2012). This choice is motivated by the locality of contrapuntal rules and their near-invariance to translation, both in time and in the frequency spectrum.

We represent the music as a stack of piano rolls encoded in a binary three-tensor $\mathbf{x} \in \{0,1\}^{I \times T \times P}$. Here I denotes the number of instruments, T the number of time steps, P the number of pitches, and $\mathbf{x}_{i,t,p} = 1$ iff the ith instrument plays pitch p at time t. We will assume each instrument plays exactly one pitch at a time, that is, $\sum_p \mathbf{x}_{i,t,p} = 1$ for all i, t.

For the present work we will restrict ourselves to the study of four-part Bach chorales as used in prior work (Allan & Williams, 2005; Boulanger-Lewandowski et al., 2012; Goel et al., 2014; Liang, 2016; Hadjeres et al., 2016). Hence we assume $I = 4$ throughout. We discretize pitch according to equal temperament, but constrain ourselves to only the range that appears in our training data (MIDI pitches 36 through 88). Time is discretized at the level of 16th notes for similar reasons. To curb memory requirements, we enforce $T = 64$ by randomly cropping the training examples.

Given a training example $\mathbf{x} \sim p(\mathbf{x})$, we present the model with the values of only a strict subset of its elements $\mathbf{x}_C = \{\mathbf{x}_{(i,t)} \mid (i,t) \in C\}$ and ask it to reconstruct its complement $\mathbf{x}_{\neg C}$. The input to the model is obtained by masking the piano rolls $\mathbf{x}$ to obtain the context $\mathbf{x}_C$ and concatenating this with the corresponding mask:

$$\mathbf{h}^0_{i,t,p} = \mathbb{1}_{(i,t)\in C}\mathbf{x}_{i,t,p} \tag{1}$$

$$\mathbf{h}^0_{I+i,t,p} = \mathbb{1}_{(i,t)\in C} \tag{2}$$

where the first dimension ranges over channels and the time and pitch dimensions are convolved over.

$$\mathbf{a}^l = \mathrm{BN}(\mathbf{W}^l * \mathbf{h}^{l-1}; \gamma^l, \beta^l) \tag{3}$$

$$\mathbf{h}^l = \mathrm{ReLU}(\mathbf{a}^l + \mathbf{h}^{l-2}) \qquad \text{for } 3 < l < L-1 \text{ and } l = 0 \bmod 2 \tag{4}$$

$$\mathbf{h}^L = \mathbf{a}^L \tag{5}$$

With the exception of the first and final layers, all of our convolutions preserve the size of the input. That is, we use "same" padding throughout and all activations $h^l, 1 < l < L$ have 128 channels. The network consists of 64 layers with 3×3 filters on each layer. After each convolution we apply batch normalization Ioffe & Szegedy (2015) (denoted by $\mathrm{BN}(\cdot)$) with statistics tied across time and pitch. After every second convolution, we introduce a skip connection from the hidden state two levels below to reap the benefits of residual learning He et al. (2015).

Finally, we obtain predictions for the pitch at each instrument/time pair:

$$p_\theta(\mathbf{x}_{i,t,p} \mid \mathbf{x}_C, C) = \frac{\exp(h^L_{i,t,p})}{\sum_p \exp(h^L_{i,t,p})} \tag{6}$$

The loss function is given by

$$\mathcal{L}(\mathbf{x}; C, \theta) = -\sum_{(i,t)\notin C} \log p_\theta(\mathbf{x}_{i,t} \mid \mathbf{x}_C, C) \tag{7}$$

$$= -\sum_{(i,t)\notin C} \sum_p \mathbf{x}_{i,t,p} \log p_\theta(\mathbf{x}_{i,t,p} \mid \mathbf{x}_C, C) \tag{8}$$

where p_θ denotes the probability under the model with parameters $\theta = \mathbf{W}^1, \gamma^1, \beta^1, \ldots, \mathbf{W}^{L-1}, \gamma^{L-1}, \beta^{L-1}$. To minimize the expected loss

$$\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})} \mathbb{E}_{C\sim p(C)} \mathcal{L}(\mathbf{x}; C, \theta) \tag{9}$$

with respect to θ, we sample piano rolls $\mathbf{x}$ from the training set and contexts $C \sim p(C)$ and optimize by stochastic gradient descent with step size determined by Adam (Kingma & Ba, 2014).

4 Relationship to NADE

Our approach is an instance of *orderless and deep* Neural Autoregressive Distribution Estimators (Uria et al., 2016). NADE models a d-variate distribution $p(\mathbf{x})$ through a factorization

$$p_\theta(\mathbf{x}) = \prod_d p_\theta(\mathbf{x}_{o_d} \mid \mathbf{x}_{o_{<d}}) \tag{10}$$

where o is a permutation, and the parameters θ are shared among the conditionals. NADE can be trained for all orderings o simultaneously using the orderless NADE (Uria et al., 2014) training procedure. This procedure relies on the observation that, thanks to parameter sharing, computing $p_\theta(\mathbf{x}_{o_{d'}} \mid \mathbf{x}_{o_{<d}})$ for all $d' \geq d$ is no more expensive than computing it only for $d' = d$. Hence for a given o and d we can simultaneously obtain partial losses for all orderings that agree with o up to d:

$$\mathcal{L}_{\text{NADE}}(\mathbf{x}; o_{<d}, \theta) = -\sum_{o_d} \log p_\theta(\mathbf{x}_{o_d} \mid \mathbf{x}_{o_{<d}}, o_{<d}, o_d) \tag{11}$$

$$\tag{12}$$

Letting $o_{<d} = C$, we obtain our loss from Equation 7

$$\mathcal{L}_{\text{COCONET}}(\mathbf{x}; C, \theta) = -\sum_{(i,t)\notin C} \log p_\theta(\mathbf{x}_{i,t} \mid \mathbf{x}_C, C) \tag{13}$$

For any one sample $(\mathbf{x}, C)$, this loss consists of $|\neg C|$ terms of the form $\log p_\theta(\mathbf{x}_{i,t} \mid \mathbf{x}_C, C)$. We let $p(C)$ be uniform in the size of the mask $|C|$ and reweight the sample losses according to

$$\tilde{\mathcal{L}}(\mathbf{x}; C, \theta) = \frac{1}{|\neg C|} \mathcal{L}(\mathbf{x}; C, \theta). \tag{14}$$

This correction, due to Uria et al. (2014), ensures consistent estimation of the negative log-likelihood of the joint $p_\theta(\mathbf{x})$.

However, we might wish to increase the difficulty by choosing $p(C)$ so as to frequently mask out large contiguous regions, as otherwise the model might learn only superficial local relationships. This is discussed in Pathak et al. (2016) for the case of images, where a model might learn only that pixels are similar to their neighbors. Similar low-level relationships hold in our case, as our piano roll representation is binary and very sparse. For instance, if we mask out only a single sixteenth step in the middle of a long-held note, reconstructing the masked out step does not require any deep understanding of music. To this end we also consider choosing the context C by independent Bernoulli samples, such that each variable has a low probability of being included in the context.

5 SAMPLING

We can sample from the model using the NADE ancestral ordering procedure. However, we find that this yields poor samples, and we propose instead to use Gibbs sampling.

5.1 NADE SAMPLING

To sample according to NADE, we start with an empty (zero everywhere) piano roll $\mathbf{x}^0$ and context C^0 and populate them iteratively by the following process. We feed the piano roll $\mathbf{x}^s$ and context C^s into the model to obtain a set of categorical distributions $p_\theta(\mathbf{x}_{i,t}|\mathbf{x}^s_{C^s}, C^s)$ for $(i,t) \notin C^s$. As the $\mathbf{x}_{i,t}$ are not conditionally independent, we cannot simply sample from these distributions independently. However, if we sample from one of them, we can compute new conditional distributions for the others. Hence we randomly choose one $(i,t)^{s+1}$ to sample from, and let $\mathbf{x}^{s+1}_{i,t}$ equal the one-hot realization. Augment the context with $C^{s+1} = C^s \cup (i,t)$ and repeat until the piano roll is populated. This procedure is easily generalized to tasks such as melody harmonization and partial score completion by starting with a nonempty piano roll.

Unfortunately, samples thus generated are of low quality, which we surmise is due to accumulation of errors. While the model provides conditionals $p_\theta(\mathbf{x}_{i,t}|\mathbf{x}_C, C)$ for all $(i,t) \notin C$, some of these conditionals may be better modeled than others. We suspect in particular those conditionals used early on in the procedure, for which the context C consists of very few variables. Moreover, although the model is trained to be order-agnostic, different orderings invoke different distributions, which is another indication that some conditionals are poorly learned. We test this hypothesis in Section 6.2.

5.2 GIBBS SAMPLING

To remedy this, we allow the model to revisit its choices: we repeatedly mask out some part of the piano roll and then repopulate it. This is a form of blocked Gibbs sampling (Liu, 1994). Blocked sampling is crucial for mixing, as the high temporal resolution of our representation causes strong correlations between consecutive notes. For instance, without blocked sampling, it would take many steps to snap out of a long-held note. Similar observations hold for the Ising model from statistical mechanics, leading to the development of the Swendsen-Wang algorithm (Swendsen & Wang, 1987) in which large clusters of variables are resampled at once.

We consider two strategies for resampling a given block of variables: *ancestral* sampling and *independent* sampling. Ancestral sampling invokes the orderless NADE sampling procedure described in Section 5.1 on the masked-out portion of the piano roll. Independent sampling simply treats the masked-out variables $\mathbf{x}_{\neg C}$ as independent given the context $\mathbf{x}_C$.

Using independent blocked Gibbs to sample from a NADE model has been studied by Yao et al. (2014), who propose to use an annealed masking probability given by

$$\alpha_n = \max\left(\alpha_{\min}, \alpha_{\max} - \frac{n}{\eta N}(\alpha_{\max} - \alpha_{\min})\right)$$

for some minimum and maximum probabilities $\alpha_{\min}, \alpha_{\max}$, number of Gibbs steps N and fraction η of time spent before settling onto the minimum probability $\alpha_{\min}$. This scheme ensures the Gibbs

Table 1: Negative log-likelihood on the test set for the Bach corpus. As discussed in the text, our numbers are not directly comparable to those of other authors due to the use of different splits. Results from Boulanger-Lewandowski et al. (2012) were based on an eighth-note temporal resolution (our resolution is sixteenth notes). Please note that our results are preliminary *validation* likelihoods.

Model	Notewise NLL	Framewise NLL
Bachbot (Liang, 2016)	0.477	–
NADE (Boulanger-Lewandowski et al., 2012)	–	7.19
RNN-RBM (Boulanger-Lewandowski et al., 2012)	–	6.27
RNN-NADE (Boulanger-Lewandowski et al., 2012)	–	5.56
COCONET, i.i.d Bernoulli(0.50)	0.924	∞
COCONET, i.i.d Bernoulli(0.25)	0.655	4.48
COCONET, i.i.d Bernoulli(0.10)	0.812	4.66
COCONET, importance sampling	0.569	3.73

process with independent resampling produces samples from the model distribution $p_\theta(\mathbf{x})$. Initially, when the masking probability is high, the chain mixes fast but samples are poor due to independent sampling. As the masking probability reduces, fewer variables are sampled at a time, until finally variables are sampled one at a time and conditioned on all the others.

Yao et al. (2014) treat independent blocked Gibbs as a cheap approximation to ancestral sampling. Indeed, per Gibbs step, independent sampling requires only a single model evaluation, whereas ancestral sampling requires as many model evaluations as there are variables to sample. Moreover, we find that independent blocked Gibbs sampling in fact yields *better* samples than the NADE procedure from Section 5.1. Samples can be heard here: `https://soundcloud.com/czhuang/sets/coconet-nade` and `https://soundcloud.com/czhuang/sets/coconet-independent-gibbs`.

6 EVALUATION

We evaluate our approach on a corpus of four-part Bach chorales. The literature features many variants of this dataset (Allan & Williams, 2005; Boulanger-Lewandowski et al., 2012; Liang, 2016; Hadjeres et al., 2016), and we follow the unfortunate tradition of introducing our own adaptation. Although this complicates comparisons against earlier work, we feel justified in doing so as our approach requires instruments to be separated, and other authors' eighth-note temporal resolution is too coarse to accurately convey counterpoint.

We rebuilt our dataset from the Bach chorale musicXML scores readily available through (Cuthbert & Ariza, 2010), which was also the basis for the dataset used in (Liang, 2016). The scores included 357 four-part Bach chorales. We excluded scores that included note durations less than sixteenth notes, resulting in 354 pieces. These pieces were split into train/valid/test in 60/20/20% ratios.

We compare with Liang (2016) based on note-level likelihood and Boulanger-Lewandowski et al. (2012) based on frame-level likelihood. Note that train/valid/test differs among both prior work and also with our work, and that Liang (2016) uses a 80/10/10% split instead.

However, evaluation of generative models is hard (Theis et al., 2015). The gold standard for evaluation is qualitative comparison by humans, and we therefore report human evaluation results.

6.1 EVALUATING LOG-LIKELIHOOD

To estimate the log-likelihood of a datapoint $\mathbf{x}$, we follow the orderless NADE approach. That is, we sample a random ordering $(i_1, t_1), (i_2, t_2), \ldots (i_{IT}, t_{IT})$, and compute the notewise log-likelihood

$$\log \hat{p}_\theta(\mathbf{x}) = \frac{1}{IT} \sum_{d=1}^{IT} \log p_\theta(\mathbf{x}_{i_d,t_d} \mid \mathbf{x}_{C_{d-1}}, C_{d-1}) \tag{15}$$

where $C_d = \bigcup_{c=1}^{d} \{(i_c, t_c)\}$. Note that we randomly crop each datapoint to be T time steps long before processing it, as this facilitates batch processing.

We repeat this procedure k times and average across all point estimates. The numbers for our models in Table 1 were obtained with $k = 5$.

The process for computing the notewise log-likelihood is akin to teacher-forcing, where at each step of the way the model observes the ground truth for all its previous predictions. To compute the framewise log-likelihood, we instead let the model run free within each frame t. This results in a more representative measure of the model's quality as it is sensitive to accumulation of error.

Table 1 lists notewise and framewise likelihoods of the validation data under variants of our model, as well as comparable results from other authors. We include four variants of COCONET that differ in the choice of the distribution $p(C)$ over contexts during training. By *importance sampling* we refer to the orderless NADE strategy discussed in Section 4, in which $p(C)$ is uniform over $|C|$ and the sampled losses are reweighted by $1/|\neg C|$. We also evaluate three variants where the contexts are chosen by biased coin flips, that is, $\Pr((i,t) \in C) = \rho$, for $\rho \in 0.5, 0.25, 0.1$. The framewise log-likelihood for $\rho = 0.5$ is listed as ∞ as its estimation repeatedly overflowed.

Overall, COCONET seems to underperform in terms of notewise likelihood, yet perform well in terms of framewise likelihood. Estimating the loss by importance sampling appears to work significantly better than determining the context using independent Bernoulli variables, as one might expect. However, the choice of Bernoulli probability ρ strongly affects the resulting loss, which suggests that some of the conditionals benefit from more training.

Table 2: Mean ($\pm$ SEM) negative log-likelihood under the model of unconditioned samples generated from the model by various procedures.

Sampling scheme	**Notewise NLL**	**Framewise NLL**
Ancestral Gibbs, $\rho = 0.00$ (NADE)	0.565 ± 0.011	3.872 ± 0.052
Ancestral Gibbs, $\rho = 0.01$	0.560 ± 0.010	3.824 ± 0.052
Ancestral Gibbs, $\rho = 0.25$	0.444 ± 0.008	3.276 ± 0.036
Ancestral Gibbs, $\rho = 0.50$	0.438 ± 0.007	3.332 ± 0.040
Contiguous Gibbs, $\rho = 0.50$	0.447 ± 0.008	3.476 ± 0.048
Independent Gibbs (Yao et al., 2014)	0.440 ± 0.008	3.348 ± 0.040

6.2 SAMPLE QUALITY

In Section 5 we conjectured that the low quality of NADE samples is due to poorly modeled conditionals $p_\theta(\mathbf{x}_{i,t} \mid \mathbf{x}_C, C)$ where C is small. We test this hypothesis by evaluating the likelihood under the model of samples generated by the ancestral blocked Gibbs procedure with C chosen according to independent Bernoulli variables. When we set the inclusion probability ρ to 0, we obtain NADE. Increasing ρ increases the expected context size $|C|$, which should yield better samples if our hypothesis is true. The results shown in Table 2 confirm that this is the case. For these experiments, we used sample length $T = 32$ time steps and number of Gibbs steps $N = 100$.

Figure 2 shows the convergence behavior of the various Gibbs procedures, averaged over 100 runs. We see that for low values of ρ (small C), the chains hardly make progress beyond NADE in terms of likelihood. Higher values of ρ (large C) enable the model to bootstrap and reach significantly better likelihood. However, high values of ρ cause the chain to mix slowly, as can be seen in the case where $\rho = 0.50$. For comparison, we included a variant, $\mathrm{Contiguous}(0.50)$, that always masks out in contiguous chunks of at least four sixteenth notes. This variant converges much more rapidly than $\mathrm{Bernoulli}(0.50)$ despite masking out equally many variables on average. Note that whereas ancestral sampling (NADE) requires $O(IT)$ model evaluations and ancestral Gibbs requires $O(ITN)$ model evaluations, independent Gibbs requires only $O(N)$ model evaluations, with typically $N < IT$.

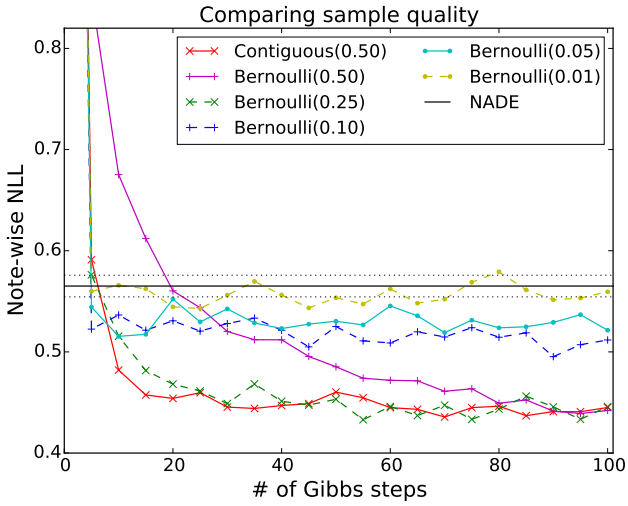

Figure 2: Likelihood under the model of Gibbs samples obtained with various context distributions $p(C)$. NADE (equivalent to $\mathrm{Bernoulli}(0.00)$) is included for reference.

6.3 HUMAN EVALUATIONS

We carried out a listening test on Amazon's Mechanical Turk (MTurk) to compare quality of samples from different sources (sampling schemes and Bach). The sampling schemes under study are ancestral Gibbs with $\mathrm{Bernoulli}(0.00)$ masking (NADE), independent Gibbs (Yao et al., 2014), and ancestral Gibbs with $\mathrm{Contiguous}(0.50)$. For each scheme, we generate four unconditioned samples from empty piano rolls. For Bach, we randomly crop four fragments from the chorale validation set. We thus obtain four sets of four sounds each. All fragments are two measures long, and last twelve seconds after synthesis.

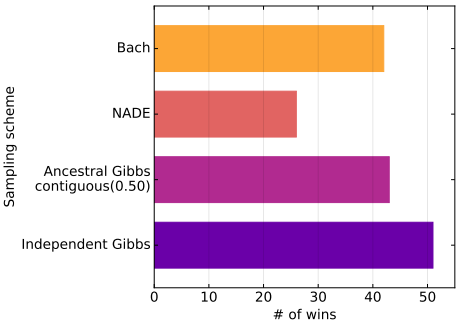

Figure 3: Human evaluations from MTurk on comparing sampling schemes.

For each MTurk hit, users are asked to rate on a Likert scale which of two random samples they perceive as more musical. The study resulted in 192 ratings, where each source was involved in 92 pairwise comparisons. Figure 6.3 reports for each source the number of times it was rated as more musical. We see that although ancestral sampling on NADE performs poorly compared to Bach, both ancestral and independent Gibbs Yao et al. (2014) were considered at least as musical as fragments from Bach, with independent Gibbs Yao et al. (2014) outperforming ancestral sampling (NADE) by a large margin. Pairwise comparisons are listed in Appendix A.

7 CONCLUSION

We introduced a convolutional approach to modeling musical scores based on the NADE (Uria et al., 2016) framework. Our experiments show that the NADE ancestral sampling procedure yields poor samples for our domain, which we have argued is because some conditionals are not captured well by the model. We have shown that sample quality improves significantly when we use blocked Gibbs sampling to iteratively rewrite parts of the score. Moreover, annealed independent blocked Gibbs sampling as proposed by Yao et al. (2014) is not only faster but in fact produces better samples.

ACKNOWLEDGMENTS

We thank Kyle Kastner and Guillaume Alain, Curtis (Fjord) Hawthorne, the Google Brain Magenta team, as well as Jason Freidenfelds for helpful feedback, discussions, suggestions and support.

REFERENCES

Moray Allan and Christopher KI Williams. Harmonising chorales by probabilistic inference. *Advances in neural information processing systems*, 17:25–32, 2005.

Amjad Almahairi, Nicolas Ballas, Tim Cooijmans, Yin Zheng, Hugo Larochelle, and Aaron Courville. Dynamic capacity networks. *arXiv preprint arXiv:1511.07838*, 2015.

Leonard E Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state markov chains. *The annals of mathematical statistics*, 37(6):1554–1563, 1966.

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

Alex J. Champandard. Neural doodle, 2016. URL `https://github.com/alexjc/neural-doodle`.

Michael Scott Cuthbert and Christopher Ariza. music21: A toolkit for computer-aided musicology and symbolic music data. 2010.

Vincent Dumoulin, Johnathon Shlens, and Manjunath Kudlur. A learned representation for artistic style. *arXiv preprint arXiv:1610.07629*, 2016.

Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.

Kratarth Goel, Raunaq Vohra, and JK Sahoo. Polyphonic music generation by modeling temporal dependencies using a rnn-dbn. In *International Conference on Artificial Neural Networks*, pp. 217–224. Springer, 2014.

Gaëtan Hadjeres, Jason Sakellariou, and François Pachet. Style imitation and chord invention in polyphonic music with exponential families. *arXiv preprint arXiv:1609.05152*, 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

Alex Lamb, Vincent Dumoulin, and Aaron Courville. Discriminative regularization for generative models. *arXiv preprint arXiv:1602.03220*, 2016.

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

Jun S Liu. The collapsed gibbs sampler in bayesian computations with applications to a gene regulation problem. *Journal of the American Statistical Association*, 89(427):958–966, 1994.

MIDI. Midi tuning standard. https://en.wikipedia.org/wiki/MIDI_Tuning_Standard. Accessed: 2016-11-12.

Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks, 2015. URL https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html.

Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. *arXiv preprint arXiv:1604.07379*, 2016.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Cognitive modeling*, 5(3):1, 1988.

Paul Smolensky. Information processing in dynamical systems: Foundations of harmony theory. Technical report, DTIC Document, 1986.

Robert H Swendsen and Jian-Sheng Wang. Nonuniversal critical dynamics in monte carlo simulations. *Physical review letters*, 58(2):86, 1987.

Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015.

Benigno Uria, Iain Murray, and Hugo Larochelle. A deep and tractable density estimator. In *ICML*, pp. 467–475, 2014.

Benigno Uria, Marc-Alexandre Côté, Karol Gregor, Iain Murray, and Hugo Larochelle. Neural autoregressive distribution estimation. *arXiv preprint arXiv:1605.02226*, 2016.

Li Yao, Sherjil Ozair, Kyunghyun Cho, and Yoshua Bengio. On the equivalence between deep nade and generative stochastic networks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 322–336. Springer, 2014.

A PAIRWISE HUMAN EVALUATION RESULTS

This appendix supplements Section 6.3 on the evaluation of samples by human subjects. Figure 3 lists the number of wins, ties and losses for each sample source against each other sample source. All pairs of sources were compared 32 times.

	I	C	N	B
I		11	20	20
C	15		12	16
N	5	14		7
B	10	10	22	

(a) Wins

	I	C	N	B
I		6	7	2
C	6		6	6
N	7	6		3
B	2	6	3	

(b) Ties

	I	C	N	B
I		15	5	10
C	11		14	10
N	20	12		22
B	20	16	7	

(c) Losses

Figure 3: Pairwise human evaluation results. Each element of Table 3(a) shows the number of times the source corresponding to the row was preferred over the source corresponding to the column. Table 3(b) shows the number of ties. Table 3(c) shows the number of losses and is the transpose of Table 3(a). Source legend: I denotes Independent Gibbs (Yao et al., 2014), C denotes Contiguous Gibbs, N denotes NADE and B denotes Bach.