EXPLORING SAMPLING TECHNIQUES FOR GENERATING MELODIES WITH A TRANSFORMER LANGUAGE MODEL

Mathias Rose Bjare[1] **Stefan Lattner**[2] **Gerhard Widmer** [1,3]
[1] Institute of Computational Perception, Johannes Kepler University Linz, Austria
[2] Sony Computer Science Laboratories (CSL), Paris, France
[3] LIT AI Lab, Linz Institute of Technology, Austria
mathias.bjare@jku.at, stefan.lattner@sony.com, gerhard.widmer@jku.at

ABSTRACT

Research in natural language processing has demonstrated that the quality of generations from trained autoregressive language models is significantly influenced by the used sampling strategy. In this study, we investigate the impact of different sampling techniques on musical qualities such as diversity and structure. To accomplish this, we train a high-capacity transformer model on a vast collection of highly-structured Irish folk melodies and analyze the musical qualities of the samples generated using distribution truncation sampling techniques. Specifically, we use nucleus sampling, the recently proposed "typical sampling", and conventional ancestral sampling. We evaluate the effect of these sampling strategies in two scenarios: optimal circumstances with a well-calibrated model and suboptimal circumstances where we systematically degrade the model's performance. We assess the generated samples using objective and subjective evaluations. We discover that probability truncation techniques may restrict diversity and structural patterns in optimal circumstances, but may also produce more musical samples in suboptimal circumstances.

1. INTRODUCTION

In recent years, developments in natural language modelling have also accelerated the field of symbolic music generation. In this context, the musical events of a music piece are represented as a sequence of symbols or tokens from a fixed vocabulary, and the goal is to learn to generate new token sequences. At present, the autoregressive transformer model [1] is the basis of many symbolic music generation models [2–5]. In this context, a conditional distribution is learned by solving a masked self-prediction task [2–5], and generation is performed with stochastic sampling techniques, e.g., ancestral sampling, or maximization-based search techniques, e.g., beam search.

However, the choice of decoding technique has been shown to impact various qualitative features of generated samples substantially. In [6], the authors showed that generation with *nucleus sampling* yields natural language samples that are more contextualized than those from conventional sampling techniques, and samples of nucleus sampling score higher in human evaluations. More recently, the authors of [7] propose *typical sampling* and show that it reduces degenerate sample generation while exhibiting performance competitive with nucleus sampling. Typical sampling is based on the authors' finding that words in human language are *typical*. More specifically, the authors show that most words of human language are, in fact, not the most likely words (lowest information content (IC)), as measured with a language model, but rather *typical* words, i.e., they have an IC close to the conditional entropy of the language model. Typical sampling explicitly enforces this condition.

We hypothesize that a careful choice of sampling technique could also improve certain aspects of music generated using language models, particularly because in [8], it has been shown that musical events tend to be typical. However, we find that many music generation systems rely on ordinary sampling techniques. In addition, studies on the effect of sampling techniques on musical qualities are limited.

In this work, we study the structural and tonal properties of music generated with different sampling techniques applied to a high-capacity transformer model. Specifically, we measure the IC, long and short-term self-similarities and scale consistency of samples generated with conventional sampling, nucleus sampling, and typical sampling. We test the sampling techniques for a well-calibrated model and for under-calibrated models. We support our findings by performing a listening study. We conduct our experiments on *The Session* dataset [9], a large dataset of well-structured monophonic music in the established musical genre of Irish traditional music. We choose this dataset since we expect it to provide suitable conditions for training a well-calibrated model. Our findings suggest that truncation techniques can address inadequacies of models that are not well-fitted to the data.

2. BACKGROUND AND RELATED WORK

Although maximization-based techniques like beam search work well for directed language generation tasks [1] (such as machine translation and summarization), beam search has been shown to produce dull and repetitive samples for open-ended language generation tasks [2] [6], an effect that can be observed in music generation as well [10]. It is, therefore, more common to use stochastic sampling techniques [3] for open-ended generation tasks. The most obvious method is ancestral sampling, where one token at a time is sampled based on the predicted distribution, conditioned on the previously generated tokens. However, it has been shown that truncating the conditional distribution (by setting the probability of specific tokens to zero, followed by renormalising), can lead to better sample quality than the non-truncated variant. An example of distribution truncation is top-k sampling, where all but the k most probable tokens are zeroed. In [12], the authors showed that top-k sampling generates more coherent samples than the non-truncated variant. In [6], it is explained that the quality improvement of top-k sampling is caused by removing unreliably estimated low-probability tokens, and it is found that top-k sampling mitigates the problem. However, it is also shown that top-k sampling is sensitive to the distribution's entropy (see Section 3.3), making it hard to select a value of k that fits both high and low certainty conditions. As a solution, they propose *nucleus sampling* that assigns zero probability to the largest set of least probable tokens that together have a probability below a given threshold. The authors find that the samples produced using the technique are preferred by humans over other sampling techniques. Nucleus sampling has been used in music generation in [13–15], but its effects are difficult to quantify without comparisons to the non-truncated case. Although nucleus sampling mitigates the problem of poorly estimated low-probability tokens, it does not prevent generating degenerated repetitive sequences caused by low entropy distributions (see Section 3). As a solution, in [7], the authors propose *typical sampling* and show that this technique prevents degenerated sample generation.

3. ANCESTRAL SAMPLING

Let $p\left(x_t|x_{<t}\right)$ be the conditional probability of a symbol x_t given previously observed symbols $x_{<t}$ (i.e., the context) and let q be a model fitted to p, e.g., a neural network fitted via likelihood maximization. Given a model q, ancestral sampling samples one token at a time using $x_0 \sim q(\cdot), x_1 \sim q(\cdot|x_0), ..., x_t \sim q(\cdot|x_{<t})$.

[1] Generation with input sequence conditioning.

[2] Generation without input sequence conditioning.

[3] In the context of generative models, "*sampling techniques*" could refer to a multitude of aspects in the generative pipeline (e.g., Gibbs sampling in restricted Boltzmann machines [11]). In our work, "*sampling techniques*", refers to techniques for obtaining samples from a trained language model.

3.1 Distribution truncation sampling techniques

In distribution truncation, a truncated distribution $\widetilde{q}$ is obtained by zeroing the probability of a subset of tokens and renormalising the resulting distribution. Formally, $\widetilde{q}$ is defined by

$$\widetilde{q}(x_t|x_{<t}) = \begin{cases} \frac{q(x_t|x_{<t})}{\sum_{v\in V} q(v|x_{<t})} & \text{if } x_t \in V \\ 0 & \text{otherwise} \end{cases}, \quad (1)$$

where V is the set of tokens with nonzero probability in $\widetilde{q}$. For the remainder of this article, we use '*conventional sampling*' to denote sampling from untruncated distributions.

3.2 Nucleus sampling

In nucleus sampling, V is defined as the smallest set such that

$$\sum_{v\in V} q(v|x_{<t}) \geq \tau, \quad (2)$$

where τ is a constant determining the number of tokens to be removed.

3.3 Typical sampling

In typical sampling [7], V is defined in terms of the token information content described below.

Definition 3.1 (Conditional information content)**.** The conditional *information content* (IC) is given by

$$IC\left(x_t|x_{<t}\right) = -\log q\left(x_t|x_{<t}\right). \quad (3)$$

In computational music perception, IC has been used to model how surprising a musical event is given the musical context [16–18].

Definition 3.2 (Conditional entropy)**.** The conditional entropy is the expected conditional information content

$$H\left(x_t|x_{<t}\right) = \mathbb{E}_{x_t\sim q(\cdot|x_{<t})}\left[IC\left(x_t|x_{<t}\right)\right]. \quad (4)$$

The entropy of a distribution explains how confident a model is. It ranges from 0 to $\log n$ where n is the number of symbols in the vocabulary, with 0 indicating that the distribution is deterministic and $\log n$ indicating that the distribution is uniformly random. In typical sampling, the probabilities of tokens with the highest deviation of information from the entropy

$$\left|H\left(x_t|x_{<t}\right) - IC\left(x_t|x_{<t}\right)\right| \quad (5)$$

are set to zero. More precisely, let $U = v_1, v_2, ..., v_n$ be an ascending ordering of the vocabulary in accordance to Equation (5). Then V is defined as the smallest prefix of U such that $q(U|x_{<t}) \geq \tau$. Equation (5) implies that V is restricted by a band around the entropy as shown in Figure 1. Therefore, also the most likely token under q can have zero probability in $\tilde{q}$. The authors of [7] note that this property, however, lowers the number of degenerately repetitive samples, as opposed to nucleus sampling, without degrading preference in human evaluations.

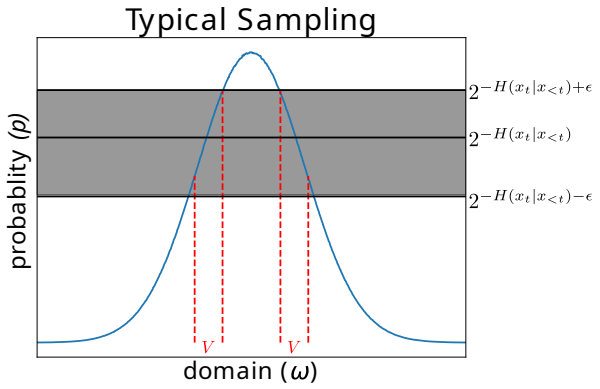


Figure 1: In typical sampling, a probability band around the entropy (dark-grey) defines the set V of tokens with non-zero probabilities in the truncated distribution.

4. EXPERIMENTS

In this section, we describe the setup for both our objective and subjective experiments, the used data, training details, degradation scenarios, and generation details, as well as objective and subjective evaluation.

4.1 Data

Our experiments are performed on monophonic symbolic music. Specifically, we use the midi-encoded version of the *The Session* dataset [9], consisting of $45,849$ traditional Irish folk tunes originally encoded in ABC notation. We discard the 5% longest sequences to lower the computational footprint of the autoregressive transformer model, and partition the dataset in training, validation, and test sets with proportions 10/12, 1/12, and 1/12, respectively. All analyses will be performed on the test set, while our generative models will be trained and optimized on the training and validation sets, respectively. The dataset contains tunes with the same name, corresponding to different versions of the same tune. We ensure that tunes with the same name appear in exactly one of the three sets.

We tokenize the sequences using a modified version of the popular REMI representation [3]. REMI serializes a score bar-wise from left to right. A bar is serialized as a sequence of tokens starting with a bar-delimiter token followed by a serialization of the notes within that bar. Each note is serialized as three tokens indicating the onset within the bar, the pitch and the duration, in that order. The position and duration tokens are quantized to 1/12th of a beat. Contrary to the original REMI implementation, we omit velocity, tempo change and chord symbols, since these are not encoded in the original ABC files either. Similar to [4], we extend the REMI representation with time-signature tokens inserted immediately after the bar token. We base our tokenization implementation on a modified version of the REMI python implementation in MidiTok [19].

4.2 Training

We train a 21-layered Transformer decoder model [20] with relative attention [2, 21] in a self-supervised prediction task. We train the model using Adam optimization [22] with a learning rate of 10^{-4} until no improvement takes place on the validation set during 10 subsequent epochs. The used batch size is 16, and the input sequence length is 512 tokens. Sequences shorter than 512 tokens are zero-padded. The negative log-likelihood (NLL) on the test dataset is measured to be $NLL = 0.30$, which is similar to the result of a recent transformer-based model trained on the same dataset [18]. We thus call this a *well-calibrated model*.

4.3 Model Degradation

In addition to the well-calibrated model, we consider two *under-calibrated models*, which we achieve by intentionally degrading the well-calibrated model. For our first degradation, we scale the logits vector h of the transformer softmax output distribution, i.e.,

$$q(x_t|x_{<t}) = \text{Softmax}(h/r), \tag{6}$$

where $r > 1.0$ is a temperature scale. This degradation increases the distribution's entropy (uncertainty) while keeping the relative ordering of the probabilities the same. Using temperature scaling, we deliberately increase the probability of token predictions x_t that fit the token context $x_{<t}$ poorly, thereby simulating the failure case of unreliably estimated tokens reported for conventional sampling (see section 2), where truncation techniques are expected to provide better results. We empirically set r to the minimal value that leads to an audible degradation of the generated sequences. This resulted in $r = 1.5$. The NLL of the test data under the temperature-degraded model is measured to be $NLL = 0.31$, which is an increase of 0.01 compared to the well-calibrated model.

Secondly, we consider an unbiased degradation where we perturb the network weights by adding a small amount of Gaussian noise. More specifically, for every weight matrix W of the well-calibrated model, we obtain a degraded weight matrix W' by adding noise z_W to W

$$W' = W + kz_W, \tag{7}$$

where $z_W \sim \mathcal{N}(0, \text{std}(W))$ and k is a constant. We sample the noise vector once and keep it fixed for all our experiments. We empirically set k to be the minimal value where sample degradations are audible, which results in $k = 0.175$. The NLL of the test data under the resulting model is measured as $NLL = 0.36$, which is an increase of 0.06.

4.4 Generation

When generating sequences with the learned models, for all models, we perform conventional sampling, nucleus sampling and typical sampling as described in section 3.

We sample until either the end-of-sequence token is encountered or a maximum length is reached. Due to computing limitations, we fix the maximum sequence length to the 80%-quantile of the dataset song-length distribution. We keep both sequences which terminate with the end-of-sequence token and sequences with the maximum length reached in our sample sets.

4.5 Objective Evaluation

The objective evaluations are performed by calculating different statistics from the generated sequences and comparing the results between different (non-)degradations, sampling types and with the original reference data.

4.5.1 Surprisal

We are interested in the degree of surprisal of the samples generated with the different sampling methods. Similar to [16–18], we measure surprisal using the IC of events. As we do not have access to the data distribution, we interpret the well-calibrated model to be an oracle that approximates the data distribution. We then use the well-calibrated model to measure the mean IC of all events from a specific sampling method and model.

4.5.2 Structural Consistency

We measure structural consistency by investigating the self-similarities of the generated pieces. Similar to [5], we compute a self-similarity distribution from samples of a given sampling method and contrast it with the similarity distribution calculated from real data. To do so, we first compute the similarity between bar pairs separated by measure lags of size t. This is done for each tune x in sample sets D according to

$$l^x_{i,i+t} = \frac{|N(i) \cap N(i+t)|}{|N(i) \cup N(i+t)|}, \tag{8}$$

where the set of notes in the i-th bar is denoted as $N(i)$, and two notes are deemed equal if their pitches, durations, and onset positions within their respective bars are identical. The similarity score $l^x_{i,j}$ between any two bars ranges from 0.0 to 1.0, with a score of 1.0 indicating that the two bars are identical. After computing the similarity for all possible lags in each tune of a sample set D, we calculate the average similarity scores of that sample set by

$$L^D_t = \frac{1}{|D|} \left(\sum_{x \in D} \sum_{j=i+t} l^x_{i,j} \right). \tag{9}$$

Note that eq. (9) does not define a probability distribution and does not, in general, sum to one. For each dataset, we then calculate an overall self-similarity score

$$SS(D) = \frac{1}{T} \sum_{t=1}^{T} L^D_t, \tag{10}$$

where T is the maximum bar lag considered. $SS(D)$ captures both short-term self-similarities, e.g., repetitions or variations of motives, and long-term self-similarities, e.g., repetitions or variations of musical segments. Similar to [5], we also consider the deviation of a sample set's similarity distribution L^D_t to the dataset's similarity distribution L_t given by

$$SE(D) = \frac{1}{T} \sum_{t=1}^{T} \left| L_t - L^D_t \right|. \tag{11}$$

We interpret this deviation as a measure of how closely the self-similarities of tunes generated with the different sampling techniques follow the self-similarities of tunes found in the dataset. We set $T = 38$ in our experiments (i.e., the smallest maximum number of bars generated by any method).

4.5.3 Tonal Consistency

We are furthermore interested in the tonality coherence of samples generated with the sampling methods. Specifically, we investigate the scale consistency [23], i.e., the maximum percentage of notes fitting a diatonic scale. The scale consistency is therefore calculated by

$$\max_{scale} \frac{\#pitch\_in\_scale(x, scale)}{\#pitches(x)}. \tag{12}$$

A scale consistency value of 1.0 indicates that all pitches are within a single scale, whereas lower values indicate more complex harmonic structures.

4.6 User Study

In addition to the objective evaluations described above, we also perform a user study to gather subjective evaluations of the tunes' musical quality, structural properties and complexity. For that, we hosted a website consisting of two pages. The first page explains the purpose of the study, specifically that it aims to evaluate sampling techniques for neural network music generation. Furthermore, the users are instructed to rate the respective tunes using the attributes *overall quality*, *short-term structure*, *long-term structure* and *complexity* using a 5-point Likert scale. The users are also asked to use appropriate headphones or loudspeakers and to announce their level of musical expertise with choices {*Beginner*, *Intermediate*, *Expert*}. On the second page, a list of 10 audio widgets is displayed, one for each tune. Below each widget, the Likert scales for the 4 different attributes (as described above) are provided for voting. In addition, the users can click on a "sheet link" that opens a window displaying the tune in staff notation. The 10 tunes for every user constitute the Cartesian product of all three sampling methods (i.e., *conventional*, *nucleus*, *typical*) and all three model modes (i.e., *well-calibrated*, *temperature degradation*, *noise-degradation*) plus a reference tune. It is ensured that every user obtains unique tunes sampled randomly from a set of 500 instances for each of the 10 types, presented in a random order. To prevent biases, every user is allowed to perform the study only once.

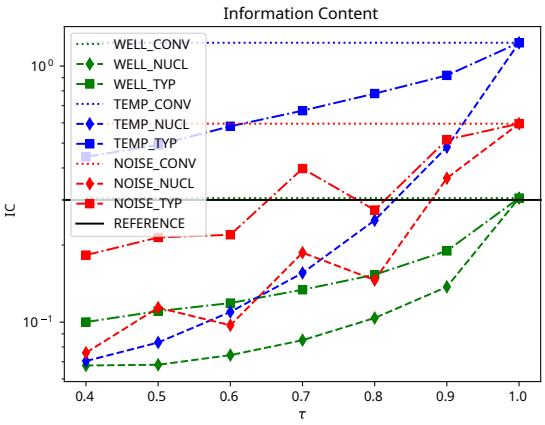


Figure 2: Information content of generated data using different sampling strategies and τ values under the well-calibrated model.

5. RESULTS AND DISCUSSION

In this section, we present the results of the experiments described in Section 4. For the figures and tables, we use the abbreviations WELL, NOISE and TEMP for the well-calibrated, noise-degraded and temperature-degraded models, respectively. To these abbreviations, we append CONV, NUCL and TYP for conventional sampling, nucleus sampling and typical sampling correspondingly.

5.1 Objective Evaluation

In the following section, we analyse and discuss the results of our objective and subjective evaluations.

5.1.1 Surprisal

We report the results of the IC estimation in Figure 2 for the truncation degrees $\tau = 0.4, ..., 1.0$. The samples from the well-calibrated model have the lowest IC and the IC of samples from the temperature-degraded model is higher than the IC of samples from the noise-degraded model. For both nucleus and typical sampling, the IC decreases with decreasing τ. For typical sampling in particular, this suggests that relatively more high information than low information tokens are pruned, similar to what is found in [8]. For most degradation scenarios and sampling methods, a τ value between 0.8 and 0.9 is shown to recover the original data distribution best.

5.1.2 Structural Consistency

We compute the self-similarity (see Equation (10)) for all models and sampling techniques and show the result in Figure 3a. Similarly, we plot the self-similarity deviation (see Equation (11)) in Figure 3b. From Figure 3a, we find that the overall self-similarity of samples produced with typical and nucleus sampling increases as τ decreases. This holds for both degraded models and the well-calibrated model. However, we find that the increase in self-similarity is more moderate for samples generated with typical sampling than those of nucleus sampling, indicating that the removal of highly probable tokens keeps the self-similarity at more moderate levels. In the temperature degradation scenario, we find that moderate levels of truncation lower the self-similarity deviation for the temperature-degraded model and thereby counteract the temperature degradation (with an optimal τ of 0.8 and 0.6 for nucleus sampling and typical sampling, respectively). In fact, in this scenario, the self-similarity of samples generated with nucleus and typical sampling follows the self-similarity of the reference distribution closer than samples generated with ordinary sampling for most tested truncation strengths. This is not the case for the unbiased noise degradation, where the self-similarity increases with higher truncation strengths, increasing also the deviation from the reference statistics.

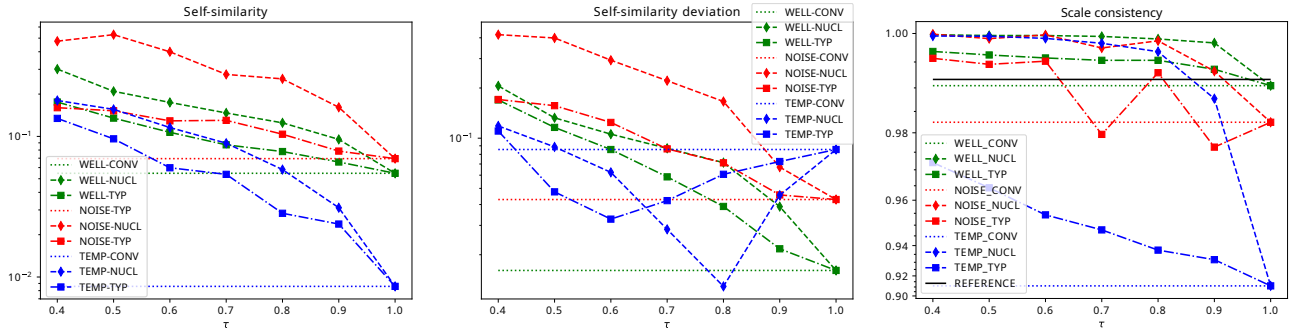
5.1.3 Tonal Consistency

We inspect the tonal consistency by calculating the scale consistency (see Equation (12)) and report the results in Figure 3c. For both nucleus and typical sampling, we find that samples generated with low values of τ lead to a higher degree of scale consistency. Furthermore, we find for any given τ that generations from typical sampling have lower scale consistency than samples generated with nucleus sampling. Especially when considering temperature degradation, the scale consistency of nucleus sampling is almost at the level of the reference distribution at $\tau = 0.9$, whereas typical sampling stays low even at high levels of τ. An important observation is that (with the exception of typical sampling in the temperature degradation scenario) there is an optimal τ for both truncation techniques that leads to a recovery of the dataset's scale consistency statistic in both degradation scenarios.

Similar to the findings in [6] for natural language, our objective evaluations in the high-temperature scenario indicate that the musical statistics of the samples generated with truncation techniques more closely match the statistics of samples from the reference distribution. This finding implies that truncation sampling techniques can be applied to music generative language models, similar to their application in natural language. This can help remove tokens with unreliable probability estimates that do not fit the musical context well. This approach may have implications for more complex datasets and limited resources, where obtaining a well-calibrated model can be challenging.

5.2 User Study

The user study was performed by 38 participants who, according to their self-assessment can be divided into 8 beginners, 18 intermediate and 12 musical experts. The presented melodies (except the *reference*) are generated as described in Section 4.4, with $\tau = 0.8$ for both, nucleus and typical sampling. Table 1 shows the user study results. As there is a high variance for all ratings, we performed for all attributes a Welch's t-test between all $m = 10$ tune types. Using a desired significance level of $\alpha = 0.05$, the corresponding Bonferroni correction to the



(a) (b) (c)

Figure 3: Structural and tonal consistency for different model degradations, sampling strategies and τ values. In (a) the self-similarity of sample sets generated with different sampling techniques is shown. Higher values indicate a higher degree of self-similarities. In (b) the deviation of the generated samples' self-similarities to the self-similarity and the data reference distribution is shown. A deviation of 0 indicates that the self-similarity of a sample set fits the reference distribution exactly. In (c) the scale consistency of different sample strategies and the reference dataset is shown.

Method	QULT	ST_STR	LT_STR	CPLX
REFERENCE	**3.7**±1.0	3.8±1.0	**3.7**±1.1	3.6±0.8
WELL_CONV	3.2±1.1	3.7±0.9	3.5±1.2	3.3±1.0
WELL_NUCL	3.6±1.1	**3.9**±1.1	**3.7**±1.1	2.8±1.0
WELL_TYP	3.4±1.2	3.6±0.9	**3.7**±1.0	3.3±1.0
NOISE_CONV	2.7±1.0	3.2±0.9	3.0±1.0	2.8±0.9
NOISE_NUCL	2.6±1.3	3.2±1.4	2.8±1.5	2.5±1.2
NOISE_TYP	2.7±1.1	3.2±1.1	3.1±1.2	2.4±1.0
TEMP_CONV	2.1±1.3	2.7±1.1	2.1±1.1	**3.7**±1.0
TEMP_NUCL	3.4±1.2	3.6±0.9	3.4±1.3	3.4±1.1
TEMP_TYP	2.2±1.1	2.7±0.9	2.4±1.0	3.3±0.8

Table 1: Results showing the mean-opinion scores of the user study $\pm$ the standard deviation. QULT denotes the overall quality estimation, ST_STR the perceived short-term structure, LT_STR the perceived long-term structure and CPLX the perceived complexity of the rated samples.

multiple comparisons problem gives a significance level of $\frac{\alpha}{\frac{1}{2}m(m-1)} = \frac{0.05}{45} = 0.001$. We can see in the first column that the human-composed reference tracks have the highest quality scores on average and that the perceived quality of the tunes tends to degrade for the noise- and temperature degradation cases as expected. The t-test shows that the users' preference for REFERENCE is significant compared to all samples of the under-calibrated models (with $p < 1 \times 10^{-4}$), except for TEMP_NUCL with $p = 0.37$. This shows that nucleus sampling can potentially improve the sample quality of low-confidence models, while typical sampling is not able to recover any degradations. Furthermore, we find that WELL_CONV, WELL_NUCL and WELL_TYP differ in QULT with $p = 0.07, 0.67$ and 0.37 respectively compared to REFERENCE. This provides some evidence that nucleus and typical sampling improves the sampling quality of well-calibrated models, but this effect is not significant. While nucleus sampling performs well in the temperature-degraded model, we observe some (non-significant) evidence of a lower complexity than conventional and typical sampling in the well-calibrated model (with $p = 0.023$ and $p = 0.044$, respectively). Typical sampling (with $\tau = 0.8$) does not cause significant differences from conventional sampling. As the p-value between NOISE_TYP and NOISE_CONV is also low (but not significant, with $p = 0.06$), there is some evidence that typical sampling slightly reduces the complexity of outputs from under-calibrated models. This could be explained by typical sampling pruning the higher and lower probability events, overall reducing the possible number of events to be sampled. The well-calibrated model performs well with all sampling techniques (no significant differences to REFERENCE), with only some non-significant evidence for lower complexity with nucleus sampling.

6. CONCLUSION

We investigated the effect of distribution truncation sampling techniques on the musical qualities of information content, self-similarity, scale consistency and complexity of samples generated under different degradation scenarios. Our objective evaluations show that a higher truncation strength leads to increased self-similarity and tonal consistency. This trend is more pronounced for samples generated with nucleus sampling compared to samples generated with typical sampling. For a well-calibrated model, we show that the increase in self-similarity and scale consistency leads to an increase in deviations of these metrics from the reference distribution. However, for under-calibrated models, we showed that the deviations from the original data statistics could often be reduced with the correct truncation strategy and carefully selected truncation levels (where a τ between 0.8 and 0.9 seems to be good trade-off value over all experiments). While nucleus sampling carries the risk to reduce complexity of the outputs, this trend could not be observed with typical sampling.

7. ACKNOWLEDGMENTS

The work leading to these results was conducted in a collaboration between JKU and Sony Computer Science Laboratories Paris under a research agreement. GW's work is supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme, grant agreement 101019375 (*"Whither Music?"*).

8. REFERENCES

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017, pp. 5998–6008.

[2] C. A. Huang, A. Vaswani, J. Uszkoreit, N. Shazeer, C. Hawthorne, A. M. Dai, M. D. Hoffman, and D. Eck, "An improved relative self-attention mechanism for transformer with application to music generation," *CoRR*, vol. abs/1809.04281, 2018.

[3] Y.-S. Huang and Y.-H. Yang, "Pop music transformer: Beat-based modeling and generation of expressive pop piano compositions," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 1180–1188.

[4] D. von Rütte, L. Biggio, Y. Kilcher, and T. Hoffman, "FIGARO: generating symbolic music with fine-grained artistic control," *CoRR*, vol. abs/2201.10936, 2022.

[5] B. Yu, P. Lu, R. Wang, W. Hu, X. Tan, W. Ye, S. Zhang, T. Qin, and T. Liu, "Museformer: Transformer with fine- and coarse-grained attention for music generation," in *NeurIPS*, 2022.

[6] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, "The curious case of neural text degeneration," in *ICLR*. OpenReview.net, 2020.

[7] C. Meister, T. Pimentel, G. Wiher, and R. Cotterell, "Typical decoding for natural language generation," *arXiv preprint arXiv:2202.00666*, 2022.

[8] M. R. Bjare and S. Lattner, "On the typicality of musical sequences," in *ISMIR (Late-breaking demo)*, 2022.

[9] B. L. Sturm, J. F. Santos, O. Ben-Tal, and I. Korshunova, "Music transcription modelling and composition using deep learning," in *Proc. Conf. Computer Simulation of Musical Creativity*, Huddersfield, UK, 2016.

[10] S. Dieleman, "Musings on typicality," 2020. [Online]. Available: https://benanne.github.io/2020/09/01/typicality.html

[11] R. Salakhutdinov, A. Mnih, and G. Hinton, "Restricted boltzmann machines for collaborative filtering," in *Proceedings of the 24th international conference on Machine learning*, 2007, pp. 791–798.

[12] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2019.

[13] W. Hsiao, J. Liu, Y. Yeh, and Y. Yang, "Compound word transformer: Learning to compose full-song music over dynamic directed hypergraphs," in *AAAI*. AAAI Press, 2021, pp. 178–186.

[14] B. Sturm and L. Casini, "Tradformer: A transformer model of traditional music," in *International Joint Conference on Artificial Intelligence*, 2022.

[15] F. Mo, X. Ji, H. Qian, and Y. Xu, "A user-customized automatic music composition system," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 640–645.

[16] L. B. Meyer, "Meaning in music and information theory," *The Journal of Aesthetics and Art Criticism*, vol. 15, no. 4, pp. 412–424, 1957. [Online]. Available: http://www.jstor.org/stable/427154

[17] M. Pearce, "The construction and evaluation of statistical models of melodic structure in music perception and composition," Ph.D. dissertation, Department of Computing, City University, London, UK, 2005.

[18] M. R. Bjare, S. Lattner, and G. Widmer, "Differentiable short-term models for efficient online learning and prediction in monophonic music," *Trans. Int. Soc. Music. Inf. Retr.*, vol. 5, no. 1, p. 190, 2022.

[19] N. Fradet, J.-P. Briot, F. Chhel, A. El Fallah Seghrouchni, and N. Gutowski, "MidiTok: A python package for MIDI file tokenization," in *ISMIR (Late-breaking demo)*, 2021.

[20] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *ICLR*, 2015.

[21] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," in *NAACL-HLT (2)*. Association for Computational Linguistics, 2018, pp. 464–468.

[22] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR (Poster)*, 2015.

[23] O. Mogren, "C-rnn-gan: A continuous recurrent neural network with adversarial training," in *Constructive Machine Learning Workshop (CML) at NIPS 2016*, 2016, p. 1.