

# Of All StrIPEs: Investigating Structure-informed Positional Encoding for Efficient Music Generation

Manvi Agarwal, *IEEE student member*, Changhong Wang, *IEEE member*, Gaël Richard, *IEEE Fellow*.

**Abstract**—While music remains a challenging domain for generative models like Transformers, a two-pronged approach has recently proved successful: inserting musically-relevant structural information into the positional encoding (PE) module and using kernel approximation techniques based on Random Fourier Features (RFF) to lower the computational cost from quadratic to linear. Yet, it is not clear how such RFF-based efficient PEs compare with those based on rotation matrices, such as Rotary Positional Encoding (RoPE). In this paper, we present a unified framework based on kernel methods to analyze both families of efficient PEs. We use this framework to develop a novel PE method called *RoPEPool*, capable of extracting causal relationships from temporal sequences. Using RFF-based PEs and rotation-based PEs, we demonstrate how seemingly disparate PEs can be jointly studied by considering the content-context interactions they induce. For empirical validation, we use a symbolic music generation task, namely, melody harmonization. We show that *RoPEPool*, combined with highly-informative structural priors, outperforms all methods.

**Index Terms**—music generation, symbolic music, transformers, positional encoding, kernels, attention, rotary positional encoding.

## I. INTRODUCTION

Generative models have been attracting increasing interest due to their impressive capability to produce high-quality, realistic samples. Among the ingredients necessary for obtaining this superior performance, two play a key role: voluminous data and ever-increasing parameter counts [1]. Domains with large-scale datasets, such as vision, text and speech, use these ingredients to obtain effective large models. In contrast, since music lacks large publicly-available datasets, it is necessary to find new ways to efficiently use existing small-scale datasets.

Another important ingredient is recent architectural advances, such as the development of Transformers and attention. Music has been no stranger to this phenomenon. However, music generated by such models often lacks essential properties that characterize human-generated ‘real’ music, such as long-term coherence and multi-resolution organization [2]. One simple way of improving music generation is to embed prior knowledge about musical structure into data-driven Transformers [3]–[5] through the positional encoding (PE) module [6]–[9], which has proven to be a good way to replace the usual (structure-free) PE without an increase in complexity.

Preprint. Manuscript submitted for review.

All authors are based at Information Processing and Communications Laboratory (LTCl), Télécom Paris, Institut Polytechnique de Paris, France.

This work was funded by the European Union (ERC, HI-Audio, 101052978). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

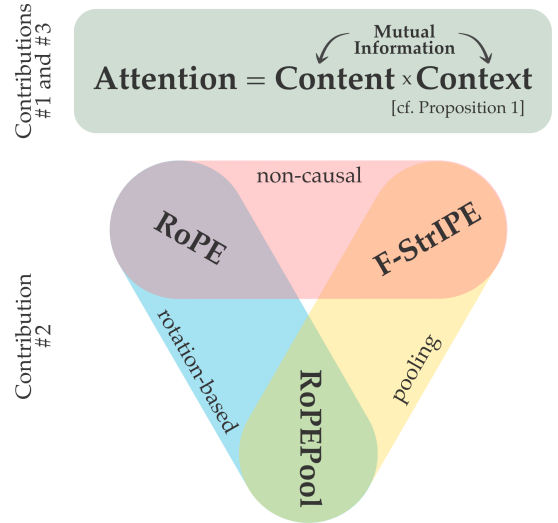


Fig. 1: Top: Efficient attention enriched with positional information can be viewed as a Tensor Product Kernel operating on context similarity and content similarity. Bottom: Relationship of the three positional encoding methods discussed in this paper (RoPE, F-StrIPE and RoPEPool)

Despite their success, Transformers have a bottleneck: their complexity scales quadratically with sequence length. This can be reduced to a linear cost by using kernel approximations [10], [11]. Recently, low-complexity Transformers were combined with musically-relevant structural information with F-StrIPE [12], which generalizes an older, structure-free PE method, Stochastic Positional Encoding [13], by using Random Fourier Features (RFF) [14]. However, an efficient PE that is conspicuously missing as a benchmark here is Rotary Positional Encoding (RoPE) [15]. Conversely, works that investigate RoPE overlook RFF-based efficient PEs, though Stochastic Positional Encoding (SPE) was the first linear-complexity relative positional encoding.

This reveals a gap in the understanding of efficient positional encoding: how can we comparatively analyze efficient PEs that are based on Random Fourier Features, such as SPE and F-StrIPE, with efficient PEs that are based on rotation matrices, such as RoPE and its variants? A cursory reading of the literature reveals that most works compare PEs largely on a task-dependent basis, with some notable exceptions that also provide insights into the characteristics of different PE methods [16]–[18]. We follow the latter line of thought, comparing PEs not just empirically through task performance, but also from a theoretical perspective, using *kernel methods*.

We present three contributions in this paper, which are summarized in Fig. 1.

*First*, we unify RFF-based PEs and rotation-based PEs into a common framework based on a canonical form of PE-enriched linear attention. We characterize this canonical form as a tensor product kernel operating on two spaces: one encoding similarity in terms of content and the other encoding similarity in terms of context. By studying the context similarity kernels produced by RoPE and F-StrIPE, we make it possible to compare the forms of attention they induce.

*Second*, we introduce a novel offshoot of RoPE, called RoPEPool, which, unlike RoPE and F-StrIPE, is capable of extracting causal relationships from temporal data. We do this by framing PE-enriched attention as feature transforms applied to queries and keys and outline the feature transforms corresponding to RoPE and F-StrIPE. We find that F-StrIPE relies on the cross-dimension independence property to use pooling on its feature transform, while RoPE approximates attention by using its feature transform without pooling. We use this insight to combine RoPE’s feature transform with the pooling operation to induce cross-dimension interactions, yielding RoPEPool. We use a synthetic dataset to illustrate the rich relationship between content similarity and context similarity in RoPEPool.

*Third*, we provide new insights to explain why rich contextual information in positional encoding leads to better performance. We probe this idea empirically with an application to symbolic music generation. In particular, we use the melody harmonization task to comparatively assess the performance of RoPE, F-StrIPE and RoPEPool on different types of contextual information. We find that RoPEPool with highly-informative structural priors performs the best on our chosen task. We make our notion of the content-context interaction precise by computing the mutual information between the content and the context under different types of contextual information. We show that performance correlates strongly with mutual information and we provide a phenomenological analysis of how the performance of different PE methods changes as context carries more information about content.

The rest of this paper is organized as follows. In Section II, we explain the background literature needed to set the stage for our work. Section III shows how different PEs can be studied using a unified framework of PE-enriched attention as well as present and analyze our novel PE method, RoPEPool. In Section IV, we introduce the experimental protocols we use for empirically validating the ideas we present in Section III. In Sections V and VI, we present the results of our empirical enquiry and show that the concepts developed in Section III are valid for real datasets.

## II. BACKGROUND

In the rest of the paper, we use  $x$  for scalars,  $\mathbf{x}$  for vectors and  $\mathbf{X}$  for matrices. Further, we use the convention “ $\mathbf{Q}/\mathbf{K}$ ” to mean “ $\mathbf{Q}$  respectively  $\mathbf{K}$ ”.

### A. Transformers, Attention and Positional Encoding

The Transformer [19] is a sequence-to-sequence architecture. Unlike recurrent neural networks, which perform se-

quential processing, Transformers process all timesteps in a sequence parallelly using a mechanism called *attention*. Given a sequence  $[\mathbf{x}_1, \dots, \mathbf{x}_T]$  as input, each element of the output sequence is computed as:

$$\mathbf{y}_m = \frac{\sum_n a_{mn} \mathbf{v}_n}{\sum_n a_{mn}} \text{ with } a_{mn} = \exp\left(\frac{a_{mn}}{\sqrt{D}}\right) \quad (1)$$

$$a_{mn} = \mathbf{q}_m \mathbf{k}_n^\top \quad (2)$$

where  $a_{mn}$  is the attention coefficient or “similarity score” for a pair of timesteps  $(m, n)$ . Inputs  $\mathbf{x}_m$  are transformed with linear layers to obtain queries  $\mathbf{q}_m$ , keys  $\mathbf{k}_m$  and values  $\mathbf{v}_m$ . Because attention does parallel processing, it lacks the notion of causality. This makes Transformers invariant to permutations in the temporal order of inputs. Hence, *positional encoding* (PE) is used to provide the model with a sense of time. Positional information can be included in two places: at the input (i.e., via  $\mathbf{x}_m$ ) or during attention computation (i.e., via  $a_{mn}$ ). Here, we focus on the latter approach, which is called Relative Positional Encoding (RPE).

### B. Efficient Attention with Positional Information

The computational cost of attention scales quadratically with sequence length. To bring this cost down to the linear regime, kernelized attention was proposed:

$$a_{mn} = \mathcal{K}(\mathbf{q}_m, \mathbf{k}_n) = \mathbb{E}\left[\phi(\mathbf{q}_m)\phi(\mathbf{k}_n)^\top\right] \quad (3)$$

where  $\mathcal{K}$  is a positive definite kernel and  $\phi(\mathbf{x}) : \mathbb{R}^D \rightarrow \mathbb{R}^{D_\phi}$  defines a *randomized feature map* for  $\mathbf{x}$  [11], [20]–[22]. Since it uses multiple instantiations,  $\phi$  captures, on average, the relationship between  $\mathbf{q}_m$  and  $\mathbf{k}_n$ , typified by  $\mathcal{K}$ . As a result, coefficients  $a_{mn}$  do not need to be computed explicitly, which produces linear-complexity Transformers.

This efficient, kernelized formulation cannot be directly used with RPE which, as introduced in [23], requires the explicit computation of attention coefficients  $a_{mn}$ . This lacuna was filled by Stochastic Positional Encoding [13], which, uses the following *canonical form of PE-enriched exact attention*:

$$a_{mn} = \sum_{d=1}^D a_{mn}^d = \left[ \sum_{d=1}^D \text{diag}(\mathbf{Q}_{:,d}) \mathbf{P}_d \text{diag}(\mathbf{K}_{:,d}) \right]_{mn} \quad (4)$$

The positional matrix  $\mathbf{P}_d$  captures the relationship between all pairs  $(m, n)$  of timesteps that come from the positional index sequences  $\mathcal{P}_Q = \{1, \dots, m, \dots, T_Q\}$ , associated with the query matrix  $\mathbf{Q}$ , and  $\mathcal{P}_K = \{1, \dots, n, \dots, T_K\}$ , associated with the key matrix  $\mathbf{K}$ . In this equation,  $\mathbf{Q}_{:,d}/\mathbf{K}_{:,d}$  extracts a  $T_Q/T_K$ -dimensional vector containing the  $d^{\text{th}}$  dimension for all timesteps of the query/key matrix. To obtain approximate attention, SPE factorizes  $\mathbf{P}_d$  as:

$$a_{mn} \approx \left[ \sum_{d=1}^D \text{diag}(\mathbf{Q}_{:,d}) \underbrace{\frac{\tilde{\mathbf{P}}_d^Q}{\sqrt{R}} \frac{\tilde{\mathbf{P}}_d^{K^\top}}{\sqrt{R}}}_{\approx \mathbf{P}_d} \text{diag}(\mathbf{K}_{:,d}^\top) \right]_{mn} \quad (5)$$

$$\tilde{\mathbf{P}}_d^{Q/K} = \frac{\Omega\left(\mathcal{P}_{Q/K}, \mathbf{f}_d, \theta_d^{Q/K}\right) \text{diag}\left(\ddot{\lambda}_d\right) \mathbf{Z}_d}{\sqrt{2N_f}} \quad (6)$$

The matrices  $\mathbf{Q}^{\text{SFF}}$  and  $\mathbf{K}^{\text{SFF}}$  combine *content* information ( $\mathbf{Q}/\mathbf{K}$ ) with *context* information ( $\tilde{\mathbf{P}}_d^Q/\tilde{\mathbf{P}}_d^K$ ), using Stochastic Fourier Features (SFF), and can be seen as PE-enriched versions of  $\mathbf{Q}$  and  $\mathbf{K}$ . The matrices  $\tilde{\mathbf{P}}_d^Q$  and  $\tilde{\mathbf{P}}_d^K$  contain representations of the positional indices associated with the queries  $\mathbf{Q}$  and the keys  $\mathbf{K}$ . As shown in (6), these matrices collect  $R$  feature realizations of index sequences  $\mathcal{P}_Q$  and  $\mathcal{P}_K$ , which provide an approximation of the positional matrix  $\mathbf{P}_d$ . More precisely, in (6), the matrix  $\Omega(\mathcal{P}_{Q/K}, \mathbf{f}_d, \boldsymbol{\theta}_d^{Q/K})$  contains sinusoidal features for the index sequence  $\mathcal{P}_{Q/K}$ , parameterized by  $N_f$  frequencies (collected in  $\mathbf{f}_d$ ) and phase shifts (collected in  $\boldsymbol{\theta}_d$ ). The diagonal matrix  $\text{diag}(\ddot{\lambda}_d)$  is used to apply gains to the sinusoidal features. The matrix  $\mathbf{Z}_d$ , consists of i.i.d. entries from a zero-mean, unit-variance Gaussian distribution. SPE goes one step beyond (5), by modifying it to:

$$a_{mn} \approx \left[ \underbrace{\left( \sum_{d=1}^D \text{diag}(\mathbf{Q}_{:,d}) \frac{\tilde{\mathbf{P}}_d^Q}{\sqrt{R}} \right)}_{\mathbf{Q}_c^{\text{SFF}}} \underbrace{\left( \sum_{d=1}^D \text{diag}(\mathbf{K}_{:,d}) \frac{\tilde{\mathbf{P}}_d^K}{\sqrt{R}} \right)}_{\mathbf{K}_c^{\text{SFF}}} \right]_{mn}^{\top} \quad (7)$$

The assumption made here, as noted earlier [13], is that the cross-dimension terms in (5) are negligible, which allows the equivalence from (5) to (7) to hold. SPE applies the feature maps  $\phi$ , used to approximate attention (3), to  $\mathbf{Q}_c^{\text{SFF}}$  and  $\mathbf{K}_c^{\text{SFF}}$  instead of  $\mathbf{Q}$  and  $\mathbf{K}$ . Hence, SPE applies a *pooling* operation to the PE-enriched representations of  $\mathbf{Q}$  and  $\mathbf{K}$  *before* applying the attention feature maps  $\phi$  to them.

### C. Rotary Positional Encoding

Parallel to the developments around SPE and F-StrIPE, another efficient positional encoding method, called Rotary Positional Encoding (RoPE) [15], has emerged in recent times. As its name suggests, RoPE is based on rotation matrices, unlike SPE and F-StrIPE, which are based on RFF. More precisely, RoPE approximates PE-enriched attention by applying rotation matrices  $\hat{\mathbf{P}}_m^Q$  and  $\hat{\mathbf{P}}_n^K$  to query and key vectors  $\mathbf{q}_m$  and  $\mathbf{k}_n$ :

$$a_{mn} \approx \mathbf{q}_m (\hat{\mathbf{P}}_m^Q)^\top \hat{\mathbf{P}}_n^K \mathbf{k}_n^\top \quad (8)$$

such that the rotations are parameterized by positions  $m$  (for the query) and  $n$  (for the key). The rotation matrices take a block diagonal form, given as:

$$\hat{\mathbf{P}}_j^A = \begin{bmatrix} \mathbf{R}_j^A(1) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \mathbf{R}_j^A(\frac{D}{2}) \end{bmatrix} \quad (9)$$

where each  $\mathbf{R}$  is a  $2 \times 2$  rotation matrix, given as:

$$\hat{\mathbf{R}}_j^A(i) = \begin{bmatrix} \cos(f_i \mathcal{P}_A[j]) & -\sin(f_i \mathcal{P}_A[j]) \\ \sin(f_i \mathcal{P}_A[j]) & \cos(f_i \mathcal{P}_A[j]) \end{bmatrix} \quad (10)$$

As hinted in the introduction, RoPE has become tremendously popular, having been deployed in several Large Language Models (LLMs), such as GPT-NeoX [24], Gemma [25],

the LLaMa model family [26] and the PaLM model family [27]. With its integration into the PyTorch ecosystem, for example, it has become the PE method of choice for machine learning practitioners working with Transformers. Several works have also dissected and improved on the properties of RoPE, such as its length generalization abilities [28]–[31]. This line of work has led to insights on its advantages, particularly from a mechanistic interpretability perspective [32].

## III. METHODS

### A. F-StrIPE: Fast, Structure-informed Positional Encoding

SPE can be directly connected to kernel methods by noting that SPE is a noisy version of Random Fourier Features (RFF) [14]. The noise-free version of SPE, which directly uses RFF to encode musically-relevant structural information in positional encoding, is called F-StrIPE [12].

F-StrIPE reformulates SPE by making two changes. First, it reconceptualizes positional encoding as a flexible way to introduce domain-specific prior knowledge about the underlying data domain of interest. As can be seen in SPE (Section II-B), positional encoding depends on a sequence  $\mathcal{P} = [p_1, p_2, \dots, p_T]$  of positional indices. Instead of using time information, with  $p_i = i$ , as is usually done in standard PE, F-StrIPE allows the use of multi-resolution structural labels, with  $p_i = \mathbf{s}(i)$ , where  $\mathbf{s}(i)$  is a vector with  $L$  structural labels. Second, it modifies  $\tilde{\mathbf{P}}_d^{Q/K}$  by noting that the term  $\mathbf{Z}_d \mathbf{Z}_d^\top / R = \hat{\mathbf{C}}_d$  appears in the approximation of  $\mathbf{P}_d$  by using Equations (5) and (6).  $\hat{\mathbf{C}}_d$  acts as an empirical covariance matrix and its elements can be written as random variables, with  $\alpha = \frac{1}{R} \sum_{i=1}^R v_i^2$  on the diagonal and  $\beta = \frac{1}{R} \sum_{i=1}^R v_i \nu_i$  on the off-diagonals, where  $v_i$  and  $\nu_i$  are standard Gaussian random variables. F-StrIPE operates in the limit of  $R \rightarrow \infty$ , where the values of  $\alpha$  and  $\beta$  converge to  $\alpha \rightarrow 1$  and  $\beta \rightarrow 0$  and, as a consequence, the empirical covariance matrix  $\hat{\mathbf{C}}_d$  converges to the theoretical covariance matrix  $\mathbf{C}_d = \mathbf{I}_{2N_f}$ . Thus, F-StrIPE takes the ideal case of  $\mathbf{C}_d$ , where the approximation of  $\mathbf{P}_d$  simplifies to:

$$\mathbf{P}_d[m, n] \approx \frac{1}{N_f} \sum_{\omega=1}^{N_f} \Lambda_\omega \cos(f_\omega(\mathcal{P}_Q[m] - \mathcal{P}_K[n]) + \Theta_\omega) \quad (11)$$

where  $\Lambda_\omega$  is the gain contributed by the matrices  $\text{diag}(\ddot{\lambda}_d)$  and  $\Theta_\omega$  is the phase-shift contributed by  $\boldsymbol{\theta}_d^Q$  and  $\boldsymbol{\theta}_d^K$ . By adjusting the  $\frac{1}{2}$  scaling factor, coming from (6), F-StrIPE obtains an expression that contains a mean. Thus, F-StrIPE results in an approximation of  $\mathbf{P}_d$  (11) that is precisely the form used by RFF for approximating stationary, positive definite kernels, yielding a direct connection between positional encoding and kernel approximation. Using this insight, F-StrIPE redesigns the positional feature matrices of (6) as:

$$\mathbf{P}_d^{Q/K} = \Omega(\mathcal{P}_{Q/K}, \mathbf{f}_d, \boldsymbol{\theta}_d^{Q/K}) \text{diag}(\ddot{\lambda}_d) / \sqrt{N_f} \quad (12)$$

where the sinusoidal feature matrix  $\Omega$  uses structure-aware positional indices as described earlier in this section. With this, F-StrIPE can replace  $\tilde{\mathbf{P}}_d^{Q/K}$  by  $\mathbf{P}_d^{Q/K}$ , yielding  $\mathbf{Q}^{\text{RFF}}/\mathbf{K}^{\text{RFF}}$  in place of  $\mathbf{Q}_c^{\text{SFF}}/\mathbf{K}_c^{\text{SFF}}$  for (5) and  $\mathbf{Q}_c^{\text{SFF}}/\mathbf{K}_c^{\text{SFF}}$  in place of  $\mathbf{Q}_c^{\text{SFF}}/\mathbf{K}_c^{\text{SFF}}$  for (7). F-StrIPE inherits the pooling technique

from SPE (7), leading to the attention feature maps  $\phi$  being applied to  $\mathbf{Q}_c^{\text{RFF}}$  and  $\mathbf{K}_c^{\text{RFF}}$  in place of  $\mathbf{Q}$  and  $\mathbf{K}$ .

### B. Interpreting PE-enriched Attention through Kernels

At first glance, F-StrIPE and RoPE seem to be completely divergent forms of PE-enriched attention. However, as we will explain in the following sections, they can be jointly analyzed with kernel methods. It is indeed possible to reframe PE-enriched attention in the idiom of kernels by considering the canonical form of PE-enriched exact attention (4) presented in Section II-B, which gives us our first insight:

**Proposition 1:** Exact attention enriched with positional information (4) can be written as a Tensor Product Kernel [33]:

$$a_{mn}^d = \mathcal{F}(q_{md}, k_{nd}) \times \mathcal{G}_d(p_m, p_n) \quad (13)$$

where  $\mathcal{F}$  is the linear kernel  $\mathcal{F}(x, x') = x \cdot x'$  applied to  $q_{md}$  and  $k_{nd}$ , capturing *content similarity*, and  $\mathcal{G}_d$  is a kernel that depends on the form of  $\mathbf{P}_d$  and is applied to positional indices  $p_m$  and  $p_n$ , capturing *context similarity*. Changing the form and approximation of  $\mathbf{P}_d$  (4) and, consequently  $\mathcal{G}_d$  (13) changes the specific form of PE-enriched attention we admit. Thus, we can infer the properties of  $a_{mn}$  by decomposing it into its constituents  $a_{mn}^d$  and analyzing their properties, specifically by verifying the forms that  $\mathcal{F}$  and  $\mathcal{G}_d$  take. We provide a sketch proof of Proposition 1 in the supplementary material.

### C. Comparing Rotation-based and RFF-based PEs

In this section, we will describe and compare RoPE and F-StrIPE in two ways. First, we will characterize the nature of attention ( $a_{mn}$ ) induced by each of these PE methods by studying the form of  $\mathcal{G}_d$  (Proposition 1) they use. Second, we will contrast these methods by reinterpreting PE-enriched attention as feature transforms applied to queries and keys.

For our first goal, we start with a characterization of RoPE through the lens of kernels. We can derive an expression using (8) and (10 for RoPE, which is as follows:

$$a_{mn} \approx \sum_{d=1}^{D/2} \underbrace{(q_{m,2d-1}k_{n,2d-1} + q_{m,2d}k_{n,2d})}_{\mathcal{F}_1} \underbrace{\cos \Delta_d^-}_{\mathcal{G}_d^{(1)}} + \underbrace{(q_{m,2d-1}k_{n,2d} - q_{m,2d}k_{n,2d-1})}_{\mathcal{F}_2} \underbrace{\sin \Delta_d^-}_{\mathcal{G}_d^{(2)}} \quad (14)$$

where we used *pairs of dimensions* as our unit of analysis and we set  $\Delta_d^- = f_d(\mathcal{P}_Q[m] - \mathcal{P}_K[n])$ . We can see that RoPE captures content similarity using  $\mathcal{F}_1$  and  $\mathcal{F}_2$  and context similarity using  $\mathcal{G}_d^{(1)}$  and  $\mathcal{G}_d^{(2)}$ . Since  $\mathcal{F}_2$  is not symmetric in its arguments,  $\mathcal{F}_2$  is not a positive definite kernel. As a result, in the case of RoPE,  $a_{mn}^d$  and, by extension,  $a_{mn}$  are also not positive definite kernels.

To compare this with F-StrIPE, we start by noting that the approximation of attention in F-StrIPE, using multiple frequencies for each dimension  $d$  (12), can be written as  $a_{mn} \approx \sum_{d=1}^D q_{md}k_{nd}\mathbf{P}_d[m, n]$ , where we used  $\mathbf{P}_d[m, n]$  from (11). Thus, the role of RFF is to obtain an approximation of

$\mathbf{P}_d$ , using  $N_f$  frequencies for each dimension  $d = 1, \dots, D$ , which is then applied to  $q_{md}k_{nd}$ . In this way, multiple frequencies are used to approximate a PE-enriched version of the product  $q_{md}k_{nd}$ .  $N_f \gg 1$  would be necessary and desirable if we used fixed parameters for the random features [34]. However, since F-StrIPE is in the regime of learnable parameters, we can directly learn the appropriate frequency, gain and offset for each dimension  $d$ . Thus, we trade-off learning-free randomization with learning-dependent optimization. In the case of learnable frequencies, with  $N_f = 1$ , we have:

$$a_{mn} \approx \sum_{d=1}^D \underbrace{q_{md}k_{nd}}_{\mathcal{F}} \underbrace{\cos \Delta_d^-}_{\mathcal{G}_d} \quad (15)$$

where we absorbed the learnable gains  $\Lambda_d$  into the optimization of  $q_{dm}$  and  $k_{dn}$  and set the phase-shifts  $\Theta_d$  to 0 in order to facilitate the comparison with RoPE. We call this method *F-StrIPE<sub>1</sub>* because it uses a single learnable frequency for each dimension. Since both  $\mathcal{F}$  and  $\mathcal{G}$  are positive definite kernels in this case, (15) gives us a positive definite kernel for  $a_{mn}^d$  and  $a_{mn}$ .

Note that, while a generalized framework of linearized relative positional encoding has been previously proposed [35], it fails to make a link between PE-enriched attention and kernels. For Proposition 1, we used the convention of treating a dimension  $d$  as our unit of analysis, but this can easily be extended to general units of analysis, such as *collections of dimensions*, such that the sum in (4) applies to such collections instead of individual dimensions. Indeed, as we saw, this is precisely what allows us to unify the analysis of RoPE and F-StrIPE.

We can further sharpen our understanding of the differences between rotation-based and RFF-based PEs by turning to our second goal and reframing PE-enriched attention as applying feature transforms to the query and key matrices. In this way, we can study the feature transforms induced by RoPE, on the one hand, and F-StrIPE<sub>1</sub>, on the other.

We begin by returning to a fact we briefly mentioned in Section III-A: F-StrIPE<sub>1</sub> inherits the pooling operation from SPE. The implication is that, in order for F-StrIPE<sub>1</sub> to use the form in (7) for approximating attention, similar to SPE, the cross-dimension terms must be negligible. To understand whether this assumption is valid for F-StrIPE<sub>1</sub>, we regard (5) and (7) as feature transforms. For (5), we can write the attention matrix  $[\mathbf{A}]_{mn} = a_{mn}$  as:

$$\mathbf{A} \approx [\mathcal{U}(\mathbf{q}_1) \ \dots \ \mathcal{U}(\mathbf{q}_{T_Q})]^\top [\mathcal{U}(\mathbf{k}_1) \ \dots \ \mathcal{U}(\mathbf{k}_{T_K})] \quad (16)$$

where  $\mathcal{U}(\cdot)$  is an *unpooled feature transform*, applied to both queries and keys, which can be written as:

$$\mathcal{U}(\mathbf{a}_i) : \mathbf{a}_i \mapsto [h(a_{i1}) \oplus h(a_{i2}) \oplus \dots \oplus h(a_{iD})] \quad (17)$$

Here, we used  $\mathbf{a}_i$  to stand in for  $\mathbf{q}_m$  and  $\mathbf{k}_n$  and  $h$  as a function that is applied independently to each unit of analysis, which, in this case, is a single dimension. Thus, each dimension of each query/key vector is transformed using  $h(\cdot)$  and concatenated together, with  $\oplus$  being the concatenation operation, to obtain  $\mathcal{U}(\cdot)$ . The role of  $h$  is to transform PE-free queries and keys

into PE-enriched queries and keys, equivalent to the role played by  $\mathcal{G}$  with respect to  $\mathcal{F}$  (Proposition 1). In contrast to (5), for (7), we can write the attention matrix  $\mathbf{A}$  as:

$$\mathbf{A} \approx [\mathcal{C}(\mathbf{q}_1) \cdots \mathcal{C}(\mathbf{q}_{T_Q})] [\mathcal{C}(\mathbf{k}_1) \cdots \mathcal{C}(\mathbf{k}_{T_K})]^\top \quad (18)$$

where  $\mathcal{C}(\cdot)$  is a *pooled feature transform*:

$$\mathcal{C}(\mathbf{a}_i) : \mathbf{a}_i \mapsto [h(a_{i1}) + h(a_{i2}) + \cdots + h(a_{iD})] \quad (19)$$

where components are added instead of being concatenated.

We can see that  $\mathcal{U}(\mathbf{q}_m)\mathcal{U}(\mathbf{k}_n)^\top = \mathcal{C}(\mathbf{q}_m)\mathcal{C}(\mathbf{k}_n)^\top$  only when the *cross-dimension independence property* is fulfilled, i.e.,  $h(q_{md})h(k_{nd'}) \approx 0$  for  $d \neq d'$ . In other words, this is precisely when (5) and (7) are equivalent.

Naturally, whether this property is met depends strongly on the nature of  $h$ . For F-StrIPE<sub>1</sub>,  $h$  is given by:

$$h(a_{id}) = [a_{id} \cos(f_d p_i) \quad a_{id} \sin(f_d p_i)] \quad (20)$$

With some simple ideas from harmonic analysis, we can show that *the cross-dimension independence property holds for F-StrIPE<sub>1</sub>*, making it possible to use (7) with F-StrIPE<sub>1</sub>. A sketch proof is provided in the supplementary materials.

Having described the feature transform for F-StrIPE<sub>1</sub>, we turn to the feature transform corresponding to RoPE. This has already been described in the literature [15] as:

$$\mathcal{U}(\mathbf{a}_i) : \mathbf{a}_i \mapsto [h(a_{i(1:2)}) \oplus \cdots \oplus h(a_{i(D-1:D)})] \quad (21)$$

$$h(a_{i(d:d+1)}) = \begin{bmatrix} a_{i,d} \cos(f_d p_i) - a_{i,d+1} \sin(f_d p_i) \\ a_{i,d+1} \cos(f_d p_i) + a_{i,d} \sin(f_d p_i) \end{bmatrix}^\top \quad (22)$$

where  $h$  is a function that is now applied independently to pairs of dimensions  $(d, d+1)$ . Note that  $\mathcal{U}(\mathbf{q}_m)\mathcal{U}(\mathbf{k}_n)^\top$  gives us precisely the expected expression for  $a_{mn}$  with RoPE (8).

At this point, it is pertinent to observe that SPE and, by extension, F-StrIPE<sub>1</sub> use pooling before applying  $\phi$  (7) to approximate downstream linear attention. However, it is possible to directly apply  $\phi$  without pooling while maintaining linear computational complexity by using (5) through its representation as an unpooled feature transform (16). The advantage of working directly with (5) is that it can allow us to use functions  $h$  where the cross-dimension independence property is not met. The drawback is that we introduce many more dimensions downstream for approximating linear attention with  $\phi$ . In fact, in contrast with F-StrIPE<sub>1</sub>, which is able to use (7) and a pooled feature transform for approximating attention, RoPE directly works with (5) and an unpooled feature transform.

To summarize, we showed in this section that RoPE and F-StrIPE<sub>1</sub> can be comparatively understood via two perspectives:

- 1) Using kernel methods: F-StrIPE<sub>1</sub> induces a positive definite kernel for attention, whereas RoPE induces a non-positive definite kernel.
- 2) Rethinking attention as feature transforms: F-StrIPE<sub>1</sub> possesses the cross-dimension independence property, which allows pooling before applying  $\phi$ . On the other hand, RoPE does not have this property and directly uses  $\phi$  without pooling.

Perspective (2) also allows us to conclude that, in fact, it is not necessary to apply pooling in order to use downstream linear attention (7). Rather,  $\phi$  can directly be used without pooling (5), as done in RoPE.

#### D. RoPEPool: Gaining expressivity using simple operations

Since RoPE does not possess the cross-dimension independence property, additional expressivity can be obtained through the cross-dimension interactions by combining RoPE's  $h$  function (22) with a pooled feature transform (19) to compute  $a_{mn}$ . With this, we can explore how expressive functions  $h$  can be leveraged to produce rich interactions between content and context by swapping concatenation for simple operations, such as addition. We then obtain the following expression for attention, analogous to RoPE (14) and F-StrIPE<sub>1</sub> (15):

$$a_{mn} \approx \sum_{d=1}^{D/2} \left\{ (q_{m,2d-1}k_{n,2d-1} + q_{m,2d}k_{n,2d}) \cos \Delta^- + (q_{m,2d-1}k_{n,2d} - q_{m,2d}k_{n,2d-1}) \sin \Delta^- + (q_{m,2d-1}k_{n,2d-1} - q_{m,2d}k_{n,2d}) \sin \Delta^+ + (q_{m,2d-1}k_{n,2d} + q_{m,2d}k_{n,2d-1}) \cos \Delta^+ \right\} \quad (23)$$

where we used  $\Delta^\pm = f_d(\mathcal{P}_Q[m] + \mathcal{P}_K[n])$ . Similar to our analysis for RoPE and F-StrIPE<sub>1</sub>, the above equation induces a form of PE-enriched attention that is a non-pd kernel. Further, in contrast to F-StrIPE<sub>1</sub> and RoPE, (23) depends not only on the lag between positional indices but also their sum. Thus, the cross-dimension interactions, obtained by using a pooled feature transform on RoPE's  $h$ , lead to a richer contextual representation in attention by introducing *asymmetry*. Since this new method is obtained by a simple pooling operation on RoPE, we call it *RoPEPool*.

So far, we have focused on distinguishing between different PE methods in rather formal terms, for example, by locating these methods within different kernel families. In the next section, we will combine this with an interpretive approach to compare their expressivities by, first, introducing some simplifications to (14), (15) and (23) to make our comparisons clearer, and second, shedding light on the benefits of the asymmetrical context information used by RoPEPool.

#### E. A Simplified Analysis of F-StrIPE<sub>1</sub>, RoPE and RoPEPool

To study how content and context interact under different PE methods, we start by making some mild assumptions about the query vectors  $\mathbf{q}_m$  and key vectors  $\mathbf{k}_n$ . We first assume that our basic unit of analysis is 2-dimensional chunks of the query and key vectors, given as  $\mathbf{q}_{m,2d-1:2d}$  and  $\mathbf{k}_{n,2d-1:2d}$ . This does not change anything for RoPE and RoPEPool, but it unifies pairs of dimensions in F-StrIPE<sub>1</sub> (15) under a single approximation of  $\mathcal{G}_d$ . Further, we assume that each 2-dimensional chunk has a Euclidean norm of 1. Then, since we can write the coordinates of any 2D unit-norm vector  $\mathbf{v}$  as  $[\cos \psi_v, \sin \psi_v]$ , where  $\psi_v$  is the angle that  $\mathbf{v}$  makes with the X-axis, we can think of these chunks solely in terms of their angular representations.

With this, we can first rewrite (14) for RoPE as:

$$a_{mn} \approx \sum_{d=1}^{D/2} \cos((\psi_q^d - \psi_k^d) + f_d(\xi_m^d - \xi_n^d)) \quad (24)$$

Here,  $\psi_q^d$  and  $\psi_k^d$  are the angles made by  $\mathbf{q}_{m,2d-1:2d}$  and  $\mathbf{k}_{n,2d-1:2d}$  with the X-axis, whereas  $\xi_m^d$  and  $\xi_n^d$  are the

angular representations of  $\mathcal{P}_Q[m]$  and  $\mathcal{P}_K[n]$  in radians. The frequency parameter  $f_d$  varies between 0 and 1 and controls the contribution of contextual information to attention.

Similarly, we can rewrite (15) for F-StrIPE<sub>1</sub> as:

$$a_{mn} \approx \sum_{d=1}^{D/2} \cos(\psi_q^d - \psi_k^d) \cos(f_d(\xi_m^d - \xi_n^d)) \quad (25)$$

and, for RoPEPool, we can rewrite (23) as:

$$a_{mn} \approx \sum_{d=1}^{D/2} 2 \cos(\psi_q^d + f_d \xi_m^d - \frac{\pi}{4}) \cos(\psi_k^d + f_d \xi_n^d - \frac{\pi}{4}) \quad (26)$$

We give the derivations of (24), (25) and (26) in the supplementary materials. We can comment on the expressivity of PE-enriched attention by studying the relationship between content  $(\psi_q^d, \psi_k^d)$  and context  $(\xi_m^d, \xi_n^d)$  in these three equations.

RoPE (24) leads to a linear interaction between content and context. F-StrIPE<sub>1</sub> (25) separates content and context information and models their interaction through an AND gate-like mechanism. A high score is only achieved when both the content and context of the query and the key match.

What we can notice about (24) and (25) is that both RoPE and F-StrIPE<sub>1</sub> model the contribution of positional information to the input in complementary ways. RoPE does this by using a phase-shift and F-StrIPE<sub>1</sub> does this by using a gain. In contrast, RoPEPool mixes the input-based and position-based information for the query and key, instead of keeping them separate. Thus, this interaction depends not only on the difference in content and context, as was the case for RoPE (24) and F-StrIPE<sub>1</sub> (25), but also on the identity of the content and context information. As hinted earlier, this introduces asymmetry into the picture, giving a richer interaction between content and context.

To further clarify these ideas, we simulate a simple toy example with a minimal setting of  $D = 2$  for (24), (25) and (26). We devise a synthetic dataset by choosing  $N$  context angles,  $C_1, C_2, \dots, C_N$ , distributed evenly over the interval  $(0, \pi)$ . Each  $C_i$  specifies the mean of a Gaussian distribution  $\mathcal{D}_i$  that generates the samples of the  $i^{th}$  context, as shown in Fig. 2, where we set  $N = 5$ . We create a dataset of  $P$  points  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_P\}$ , where each  $\mathbf{x}_j$  is naturally associated with an angular coordinate  $\psi_j$  that has been generated from one of the  $N$  Gaussians as  $\psi_j \sim \mathcal{N}(C_k, \sigma^2)$ ,  $k \in \{1, \dots, N\}$ . Further,

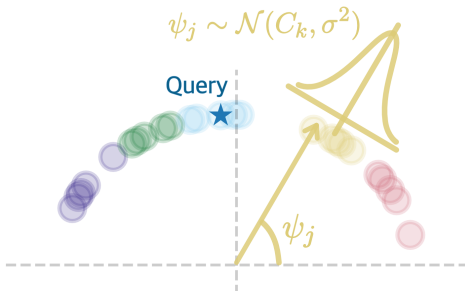


Fig. 2: Generating a toy dataset to study the characteristics of different positional encoding methods for the  $D = 2$  setting

$\mathbf{x}_j$  is also associated with  $C_k$ , the mean of the distribution it was generated from. Thus, for each point  $\mathbf{x}_j$ ,  $\psi_j$  defines its *content*, capturing low-level, high-frequency information, and  $\xi_j = C_k$  defines its *context*, capturing high-level, low-frequency information.

We designate one of the  $P$  points as our ‘query’  $\mathbf{x}_q$ , which we mark with a star symbol in Fig. 2. We are interested in measuring the score or similarity, given by  $a_{qj}$ , between  $\mathbf{x}_q$  and all the points in our dataset, using (24), (25) and (26). We show the results in Fig. 3. The query  $\mathbf{x}_q$  is marked with a star on the X-axis and we indicate which datapoints belong to the same context as  $\mathbf{x}_q$  (marked in blue; we call these *same-context keys* in our discussion below) and which datapoints belong to a different context (marked in green; we call these *different-context keys*).

We observe that F-StrIPE<sub>1</sub> exhibits attention values that are weakly modulated by  $f$  and strongly modulated by the interaction between content and context. Discriminability between same-context keys and different-context keys with respect to the query remains robust for the full range of  $f$  values. As discussed earlier, high values are achieved only when both the content and context of the key match those of the query, which can be seen clearly in Fig. 3. For RoPE, this discriminability weakens as positional information starts to have an influence on attention, i.e., as  $f$  increases. Thus, even though RoPE and F-StrIPE<sub>1</sub> share the same expressivity, increasing positional information effectively *blurs* the differences between the query and keys for RoPE, which is not the case for F-StrIPE<sub>1</sub>.

RoPEPool, on the other hand, exhibits an entirely different kind of behaviour to F-StrIPE<sub>1</sub> and RoPE. For a fixed value of  $f$ , RoPEPool potentiates the scores of the keys in one  $\psi_j$  range and depresses the scores of the keys under another  $\psi_j$  range. As  $f$  changes, these ranges also change. Thus, depending on the value of  $f$ , which we set to be learnable, RoPEPool can select a set of potentiated keys where the high scores concentrate and a set of depressed keys where the low scores concentrate. This is unlike RoPE and F-StrIPE<sub>1</sub>, where the highest scores are always concentrated in the same-context keys. Furthermore, due to the left-right asymmetry of the heatmap of RoPEPool, keys at a distance of  $-\theta$  from the query get a different score than keys at a distance of  $+\theta$ . This is starkly different from RoPE and F-StrIPE<sub>1</sub>, where both these

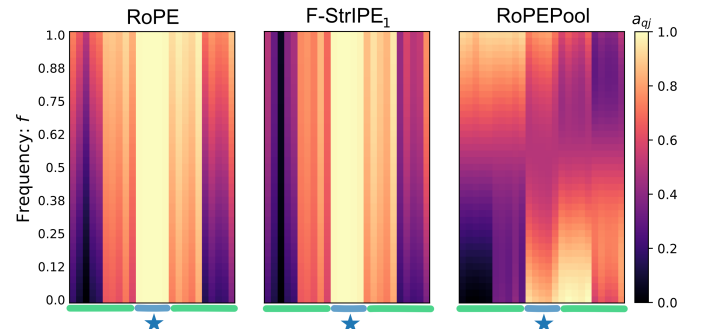


Fig. 3: Different PE methods exhibit different trade-offs between content and context



sets of keys get the same score for all values of  $f$ . This allows us to model *causality* with RoPEPool, which is a particularly salient feature in temporal sequences, such as musical signals.

Our analysis suggests that, in terms of expressivity, RoPE is the least expressive and RoPEPool is the most expressive, with F-StrIPE<sub>1</sub> occupying the middle rank. We hypothesize that RoPEPool should perform the best among these three methods on problems that require asymmetric, causal information.

#### IV. EXPERIMENTS

##### A. Dataset and Input Representation

We use the Chinese POP909 dataset [36] for our experiments. Each MIDI file in this dataset contains 3 tracks: melody, bridge and piano. The bridge acts as a second melody track, supporting the development of the themes in the melody track. The piano track contains polyphonic accompaniment to the melody and bridge tracks. Each file is also paired with fine-grained multi-resolution structural annotations [37]. For this paper, following previous findings [12], we only use chord annotations for structural information, which uses a resolution of quarter-note. We use the POP909 alignment dataset [6] to correctly sync the structural labels with the input. We work with the MIDI data in the form of binary pianorolls  $\mathbf{X} \in \mathbb{B}^{(n_{\text{tracks}} \times 128) \times n_{\text{time}}}$ ,  $\mathbb{B} = \{0, 1\}$ , where  $n_{\text{tracks}}$  is the number of tracks and  $n_{\text{time}}$  is the number of timesteps in the pianoroll. The POP909 dataset has approximately 400 unique chords. For each chord, we have access to two types of information: the scalar root note and the 12-dimensional chroma vector, which indicates the pitches that make up the chord. We investigate both token-based and vector-based representations of this information, as we will describe in later sections.

##### B. Task: Melody Harmonization

As input, the model is given the melody and bridge tracks  $[\mathbf{x}_n \in \mathbb{B}^{(n_{\text{tracks}}-1) \times 128}]$ , with  $n \in \{1, \dots, n_{\text{time}}\}$ . As output, the model produces all three tracks  $[\mathbf{y}_n \in \mathbb{B}^{n_{\text{tracks}} \times 128}]$ . Note that, here, we are in an extreme parallel decoding regime, where we require our model to produce the full accompaniment track at once and do not allow our model to autoregressively condition later predictions on earlier predictions. We train our models on sequences of length 16 bars. We test two settings: the *in-domain* setting, with sequences of length 16 bars, and the *length generalization* setting, with sequences of length 64 bars.

##### C. Model and Training

We use a 2-layer causal encoder Transformer with 4 heads and 512 model dimension. We train for 15 epochs with a batch size of 8, gradient clipping and curriculum learning [38]. We use two learning rate schedulers: a linear warmup, applied for the first three epochs, and a decay, applied for each epoch. We do a grid-search for two hyperparameters: learning rate (choices:  $\{1, 5, 10\} \times 0.0001$ ) and post-processing binarization strategy [6] (choices: thresholding, thresholding with merge). The first binarization strategy uses a fixed threshold on the logits to obtain binarized outputs. The second binarization strategy, first, uses a fixed threshold to binarize the logits and, second, allows the gap between two binarized notes to be filled when this gap is less than a minimum distance.

##### D. Types of Contextual Information

To understand how context influences model performance, we represent positional information in several ways, varying how much local versus global information the context contains.

To encode fully *local* information, we use *time* (TIME). Since each time-step is assigned a unique token, this representation assumes all inputs in a sample are independent.

To encode fully *global* information, we use *binary chroma chord vectors* (BIN), which contain ones for the notes that occur in a chord and zeros for those that don't. This accounts for the harmonic content of each chord, allowing us to identify repetitions of harmonically-similar chords across different samples at the level of the dataset.

Between these two extremities, we use two *blended* chord representations, which meld together local and global information. The first is *repetition-based chord tokens* (REP). Here, we first represent each chord as a unique token, taking into account both its root note and chroma vector. Then, to reduce the number of tokens and focus on repetitions within samples, we re-number the tokens in each sample based on their numerical order. The second is *key-matched chord tokens* (KEY). Here, we use the fact that the POP909 dataset provides the key of each song. Using this, we 'normalize' each chord with respect to its song's key, mapping the chord to its counterpart in either the C:major scale or the C:minor scale. For example, the E:minor chord in D:major scale is mapped to the D:minor chord in C:major scale. Thus, we focus on the function of each chord across samples, while blurring harmonic correlations between chords and a song's pitches.

It is worth noting that we selected three hand-designed representations, based in concepts from music theory, to simplify our experiments, but optimization-based representations [39] can also be used with the PE methods studied in this paper.

##### E. Baselines

To construct baselines for each type of contextual information described above, we investigate three variants of RoPE. The first variant, RoPE(A), is the standard method from literature [15], where the frequencies are fixed, following an exponential distribution, and all heads contain the same frequencies. The second variant, RoPE(B), modifies RoPE(A) by allowing different heads to have different frequencies. The third variant, RoPE(C), builds on RoPE(B) by allowing frequencies to be learned as parameters. For each contextual information type, we choose the RoPE variant that performs the best. Thereafter, we compare these chosen RoPE baselines with F-StrIPE<sub>1</sub> and RoPEPool for each context type. For F-StrIPE<sub>1</sub> and RoPEPool, we use a system analogous to RoPE(C), where heads have different, learnable frequencies.

##### F. Evaluation

Following existing work [6], [12], we choose four objective metrics from the literature to compare the ground truth and the generated music. Each focuses on one axis of musical quality.

We use the *Self-Similarity Matrix Distance* (SSMD) [40] to evaluate large-scale and small-scale structural properties. We

compute chroma onset vectors, giving the number of onsets per chroma per half-measure, for the target and the prediction. For each chroma onset vector, we calculate its self-similarity matrix (SSM) by taking pairwise cosine similarities between all elements. We take the absolute difference between the target’s SSM and the prediction’s SSM, giving us a matrix with entries varying between 0 and 2. The SSMD is the mean of this matrix, linearly scaled to vary between 0 and 100.

We use *Chroma Similarity (CS)* [40] to measure melodic consistency. The harmonic match between each half-measure in the target and its counterpart in the prediction is the cosine similarity between their corresponding chroma onset vectors, derived as described above. The mean of these cosine similarities, varying between -1 and 1, is linearly scaled to vary between -100 and 100, and this gives us the CS.

We use *Grooving pattern Similarity (GS)* [2] to assess rhythmic consistency. The grooving pattern of a pianoroll is a vector that encodes a 1 for the quarter-notes where onsets occur and 0 for those where no onsets occur. The XOR function is applied to the grooving patterns of the target and the prediction, after which we take the mean of the resulting vector, giving us a number between 0 and 1. We linearly scale this to vary from 0 to 100, giving us the GS.

To estimate polyphonicity, we use *Note Density Distance (NDD)* [6], [41]. For each 16<sup>th</sup>-note of the target and prediction, we calculate the total number of pitches that occur. The NDD is the average percentage of missing pitches in the prediction, with the number of pitches in the target giving us the maximum possible value. It varies between 0 and 100.

Note that all four metrics are scale-invariant to sequence length, making the results from the in-domain and length generalization scenarios comparable with each other.

## V. RESULTS

In Table II, we first present the results on the three variants of RoPE described in Section IV-E. From these, we select the RoPE baseline with the best performance for each context type and compare this with F-StrIPE<sub>1</sub> and RoPEPool. These results are shown in Table III.

For each PE method in both tables, we train models using five seeds and report the mean and standard deviation for each metric. For Table III, we underline the PE method with the best mean score for each context type. However, we check if there is a statistically significant difference between the underlined method and other methods within the same context by using two-sided t-tests [42]. For a pair of PE methods, we first use a Levene test [43] to check whether the pair of score distributions have the same variance. If this is the case, we use Student’s t-test to make the comparison, and, if not, we use Welch’s t-test [44]. We use a \* with

the underlined score to indicate that there is a significant difference ( $p < 0.05$ ) between the best (underlined) mean score and the next-best mean score. If we do not find any significant difference in this comparison, we use a † to indicate that there is a statistically significant difference between the best mean score and the third-best mean score. This is to gain a fine-grained understanding on how different PE methods perform relative to each other within a given context. In Table III, we additionally also highlight the best PE method among the different underlined PE methods in bold in order to indicate which combination of context and PE is optimal.

### A. Ordering Context Type by Information Content

Before we describe the results in Tables II and III, let us formalize the notion of ‘local versus global’ with regards to context type, which we introduced in Section IV-D. To do this, we calculate the mutual information (or MI score) between the distribution of positional information and the data distribution. We convert the binary pianorolls into an array of individual pitches and match each pitch with its corresponding positional index, which changes depending on the context type and the timestep in which the pitch occurs in the pianoroll. Then, we calculate the MI score between the input, represented as pitches, and their positions. We report this score for each context type in Table I. Intuitively, the mutual information between these two distributions measures how well we can predict one from the other. We expect that a higher MI score makes it easier for the model to learn the data distribution by leveraging contextual information via positional encoding.

### B. Comparing PE Methods within Context Types

For each context type, we first describe the results for selecting a RoPE baseline. Then, we compare the results of the chosen RoPE baseline with those of the other PE methods.

For Table II, in the in-domain setting, we see statistically significant results for two context types, namely, REP and BIN. In the length generalization setting, we see this for three context types, namely TIME, REP and BIN. For REP, RoPE(B) performs the best and, for BIN, RoPE(C) performs the best, in both settings. For TIME, RoPE(C) performs the best, particularly in the length generalization setting. Finally, for KEY, there is no statistically significant difference between the different PE methods, with RoPE(B) or RoPE(C) achieving the best mean score in most metrics for both settings.

Similar to Table II, TIME in Table III does not have statistically significant results in the in-domain scenario, but has statistically significant results in the length generalization scenario. Similarly, for KEY, F-StrIPE<sub>1</sub> and RoPEPool jointly perform best, although, unlike Table II, the difference with RoPE is statistically significant. In REP, RoPE(B) continues its win from Table II and outperforms the other PE methods. For BIN, RoPEPool performs the best, with F-StrIPE<sub>1</sub> showing similar performance with a slightly worse mean score.

### C. Utility of Structural Information

In Table III, on comparing the best PE methods from each context, we find that BIN, encoding global musically-relevant

Context type	Mutual information with data distribution (in nats)
TIME	0.0085
REP	0.0436
KEY	0.0476
BIN	1.0075

TABLE I: Mutual information between content and context



Configuration	Train = 16 bars; Test = 16 bars								Train = 16 bars; Test = 64 bars							
	CS ↑		SSMD ↓		GS ↑		NDD ↓		CS ↑		SSMD ↓		GS ↑		NDD ↓	
<b>Time</b>																
RoPE(A)	2.44	0.06	14.65	0.0	7.46	0.05	94.1	0.04	5.83	3.27	13.8	0.0	18.27	9.1	91.79	0.84
RoPE(B)	1.61	0.74	14.66	0.01	6.72	0.71	94.38	0.27	2.23	0.09	13.8	0.0	8.44	0.21	92.72	0.03
RoPE(C)	4.5	4.16	14.65	0.02	10.42	5.32	93.82	1.01	9.1	0.49 <sup>†</sup>	13.79	0.0*	26.78	1.19 <sup>†</sup>	91.26	0.2 <sup>†</sup>
<b>Repetition-based chord tokens</b>																
RoPE(A)	5.42	5.51	14.61	0.06	10.89	6.65	92.75	2.27	15.42	1.18	13.67	0.01	42.4	3.23	86.77	0.92
RoPE(B)	18.37	1.11*	14.36	0.03*	25.63	1.76*	86.13	0.57*	18.17	1.92*	13.46	0.02*	49.56	3.12*	82.9	0.54*
RoPE(C)	4.59	5.16	14.59	0.11	9.76	5.6	92.84	2.57	9.46	7.28	13.62	0.15	27.27	20.17	88.35	4.06
<b>Key-matched chord tokens</b>																
RoPE(A)	11.05	5.62	14.49	0.09	16.89	6.59	89.64	2.56	14.59	5.24	13.63	0.08	39.02	12.34	86.82	2.35
RoPE(B)	11.85	4.6	14.5	0.07	18.49	5.52	89.85	1.99	16.2	2.21	13.63	0.01	43.67	3.95	85.87	0.39
RoPE(C)	12.03	4.51	14.49	0.07	18.57	5.37	89.87	1.93	15.84	1.09	13.62	0.02	42.04	2.65	86.68	0.85
<b>Binary chroma vectors</b>																
RoPE(A)	1.81	0.34	14.64	0.01	6.76	0.35	94.3	0.14	10.04	4.21	13.74	0.02	30.39	11.72	89.44	1.69
RoPE(B)	1.9	0.18	14.66	0.01	7.07	0.18	94.29	0.07	3.51	2.56	13.8	0.0	11.93	7.12	92.31	0.73
RoPE(C)	22.14	0.56*	13.98	0.01*	26.45	0.52*	79.25	0.29*	26.27	0.44*	13.13	0.01*	66.81	0.6*	75.02	0.34*

TABLE II: Comparing different RoPE baselines (Section IV-E) using different contexts (Section IV-D)

Configuration	Train = 16 bars; Test = 16 bars								Train = 16 bars; Test = 64 bars							
	CS ↑		SSMD ↓		GS ↑		NDD ↓		CS ↑		SSMD ↓		GS ↑		NDD ↓	
<b>Time</b>																
RoPE(C)	4.5	4.16	14.65	0.02	10.42	5.32	93.82	1.01	9.1	0.49*	13.79	0.0	26.78	1.19*	91.26	0.2
F-StrIPE <sub>1</sub>	6.01	2.67	14.64	0.02	11.9	3.24	93.2	0.76	6.68	1.12	13.79	0.0	20.67	2.59	91.4	0.21
RoPEPool	2.49	2.64	14.66	0.01	7.79	3.32	94.28	0.68	7.23	0.62	13.79	0.0*	22.13	1.65	91.18	0.2
<b>Repetition-based chord tokens</b>																
RoPE(B)	18.37	1.11 <sup>†</sup>	14.36	0.03 <sup>†</sup>	25.63	1.76 <sup>†</sup>	86.13	0.57 <sup>†</sup>	18.17	1.92*	13.46	0.02*	49.56	3.12*	82.9	0.54*
F-StrIPE <sub>1</sub>	17.71	0.97	14.37	0.03	24.82	1.22	86.81	0.71	13.85	1.18	13.59	0.03	37.66	1.93	86.47	0.66
RoPEPool	14.23	1.13	14.47	0.02	20.87	1.86	88.81	0.29	11.67	0.58	13.7	0.01	34.11	1.12	88.04	0.52
<b>Key-matched chord tokens</b>																
RoPE(C)	12.03	4.51	14.49	0.07	18.57	5.37	89.87	1.93	15.84	1.09	13.62	0.02	42.04	2.65	86.68	0.85
F-StrIPE <sub>1</sub>	19.73	0.4	14.1	0.02*	24.08	0.48	81.41	0.27 <sup>†</sup>	23.6	0.51 <sup>†</sup>	13.2	0.05 <sup>†</sup>	60.87	1.28 <sup>†</sup>	77.47	0.35 <sup>†</sup>
RoPEPool	21.05	1.22 <sup>†</sup>	14.18	0.06	26.05	1.53*	82.46	0.96	23.11	1.43	13.28	0.09	59.74	2.46	78.69	1.37
<b>Binary chroma vectors</b>																
RoPE(C)	22.14	0.56	13.98	0.01	26.45	0.52	79.25	0.29	26.27	0.44	13.13	0.01	66.81	0.6	75.02	0.34
F-StrIPE <sub>1</sub>	24.91	0.35	13.73	0.03 <sup>†</sup>	28.19	0.31	77.15	0.33	28.3	0.81	12.95	0.04 <sup>†</sup>	71.35	0.99	73.97	0.48
RoPEPool	25.83	1.02 <sup>†</sup>	13.74	0.07	29.11	0.97 <sup>†</sup>	76.73	1.09 <sup>†</sup>	28.6	0.8 <sup>†</sup>	12.99	0.06	72.56	1.54 <sup>†</sup>	73.77	1.23

TABLE III: Comparing different PE methods using different contexts. We chose the best RoPE baseline from Table II to compare against F-StrIPE<sub>1</sub> and RoPEPool

information, performs the best, and TIME performs the worst, with the blended representations, namely, REP and KEY, occupying the middle ranks. In both the in-domain and length generalization scenarios, KEY enables higher performance than REP, although the difference between the two is not significant for GS in the in-domain setting ( $p > 0.5$ ).

#### D. Length Generalization

In Table III, with the exception of RoPE applied to REP, all methods and context types lead to an amelioration of scores from the in-domain scenario to the length generalization scenario, which is particularly dramatic for GS in BIN. Testing on out-of-domain samples tends to augment the score differences between various PE methods, especially for GS.

## VI. DISCUSSION

### A. Explaining the Success of Structural Positional Encoding with Mutual Information

Table I formalizes our intuition that TIME contains no information about connections within a sample, since it assumes that all time-steps are contextually independent, while REP, KEY and BIN introduce progressively stronger contextual information about the input distribution. Note that even though KEY has a higher MI score than REP, they both fall within the same order of magnitude. We contrast this with the comparison between TIME, REP/KEY and BIN, which lead to MI scores at increasing orders of magnitude. This captures the intuition that moving from local to blended to global information in context leads to fundamental changes in what additional insight can be gained with respect to the input distribution, whereas moving within the same informational regime (e.g., REP to KEY) introduces subtler interactions.

By combining our observations from Sections V-A and V-C,

we argue that this is precisely what makes task-specific priors, such as structural information, better than generic priors, such as time information, in positional encoding. This point is further strengthened by the fact that the ranking in MI scores (Table I) matches precisely the ranking in performance with respect to context (Table III). These findings could also explain how and why harnessing information about the structural properties of data has been found to aid learnability and improve performance in other generative architectures such as diffusion models, both with synthetic [45] and real data [46].

Having translated our intuitions into a quantitative form, we can now analyze the empirical results from Tables II and III in further detail while bearing these ideas in mind.

### B. A Phenomenological Perspective on Performance vis-à-vis Information Content

As presented in Section V-B, the impact of context on performance has two facets: the relative performance of different PE methods (i.e., which method performs best) and the difference between these methods (i.e., is the best performance significantly different from the others). To disentangle them, we can plot the in-domain performance of each PE method on each metric from Table III against the MI scores from Table I. The results are displayed in Fig. 4.

First, this plot, along with Table III, suggests the existence of a transition from no differentiation between PE methods, in the no-information setting (TIME), to simpler PE methods performing better than more complex PE methods, in limited-information scenarios (REP and KEY), to more complex PE methods performing better than simpler PE methods, in high-information scenarios (BIN). Thus, as contextual information in PE increases, there is also the *emergence of differentiation* between PE methods. We can see this in Fig. 4 by comparing TIME, where the different PE methods largely overlap, with other context types, where the PE methods develop distinct distributions. Second, it shows that the mutual information between data and its positional representation correlates strongly

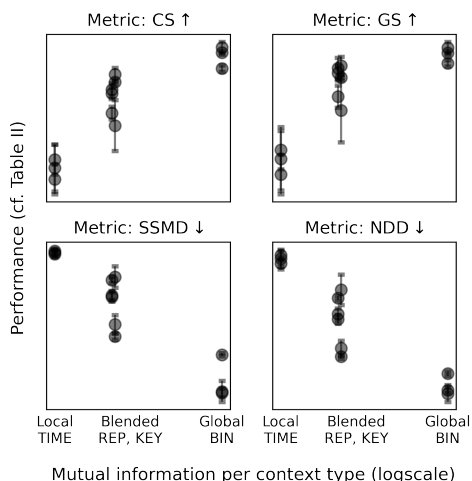


Fig. 4: Performance as a function of how much information about the data distribution is contained in the positional information; drawn from the in-domain results in Table III

with performance, adding further evidence to our analysis in Section VI-A. The more information that is encoded about the input distribution with positional encoding, the better is the performance of the model. Finally, it shows that imperfect mutual information (REP/KEY) leads to a large spread of results. This region of positional information is defined by large error bars and a wide range of mean scores.

These observations enable us to characterize the interaction between context type and comparative performance of PEs using two interrelated phenomena. First, increasing the relevance of positional information with respect to the input diversifies performance, such that differences between PEs become more evident as the context distribution starts to encode more information about the content distribution. This indicates that empirical works comparing different PE methods might see none or highly dampened differences in the performance of different PEs, not due to the PE’s inherent properties, but because the PE does not encode enough information, for example, in the typical case where TIME is used. Second, there exists an intermediary stage of relevance, with imperfect mutual information between the positions and the input, that exhibits a large variance in performance among the different PE methods. This indicates that, under imperfect structural information, performance is highly initialization-dependent and optimization in such cases might be unstable. This might also explain why, for example, on REP, RoPE(B) does better than RoPE(C) in Table II and better than F-StrIPE<sub>1</sub> and RoPEPool in Table III.

## VII. CONCLUSION

In this paper, we compared two families of efficient positional encoding methods: those based on Random Fourier Features and those based on rotation matrices. We began by unifying these families under a common framework of PE-enriched attention, which we also used to develop a novel PE called RoPEPool. With a toy example, we studied the expressiveness of RoPE, F-StrIPE<sub>1</sub> and RoPEPool and showed that, unlike RoPE and F-StrIPE<sub>1</sub>, RoPEPool can extract causal relationships from temporal sequences. To assess the validity of our ideas, we used melody harmonization, a symbolic music generation task. We also varied the type and representation of positional information used, ordering them by how much they tell us about the structure of the data. We found that combining RoPEPool with highly-informative structural information performs the best, matching our conclusions from the toy example. We went beyond these task-specific findings to clarify the role of information about data structure in PE. We used the mutual information between the content distribution (via the input) and the context distribution (via the PE) to explain why rich contexts aid learnability. We provided a phenomenological account of how the performance of different PEs varies as context information increases, enriching our comparative study.

## ACKNOWLEDGEMENTS

We thank Emmanouil Benetos for helpful discussions around variations of chord representations.

## REFERENCES

- [1] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," *arXiv*, 2020. [Online]. Available: <https://arxiv.org/abs/2001.08361>
- [2] S. Wu and Y. Yang, "The Jazz Transformer on the Front Line: Exploring the Shortcomings of AI-composed Music through Quantitative Measures," *Conference of the International Society for Music Information Retrieval (ISMIR)*, 2020.
- [3] S. Ji, X. Yang, and J. Luo, "A survey on deep learning for symbolic music generation: Representations, algorithms, evaluations, and challenges," *ACM Computing Surveys*, vol. 56, no. 1, 2023.
- [4] G. Richard, V. Lostanlen, Y.-H. Yang, and M. Miller, "Model-based deep learning for music information research: Leveraging diverse knowledge sources to enhance explainability, controllability, and resource efficiency," *IEEE Signal Processing Magazine*, vol. 41, no. 6, pp. 51–59, 2024.
- [5] K. Bhandari and S. Colton, "Motifs, Phrases, and Beyond: The Modelling of Structure in Symbolic Music Generation," *International Conference on Computational Intelligence in Music, Sound, Art and Design*, 2024.
- [6] M. Agarwal, C. Wang, and G. Richard, "Structure-Informed Positional Encoding for Music Generation," *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024.
- [7] Y. Ren, J. He, X. Tan, T. Qin, Z. Zhao, and T.-Y. Liu, "PopMAG: Pop Music Accompaniment Generation," *ACM International Conference on Multimedia (MM)*, 2020.
- [8] Z. Guo, J. Kang, and D. Herremans, "A Domain-Knowledge-Inspired Music Embedding Space and a Novel Attention Mechanism for Symbolic Music Modeling," *AAAI Conference on Artificial Intelligence (AAAI)*, 2023.
- [9] J. Liu, Y. Dong, Z. Cheng, X. Zhang, X. Li, F. Yu, and M. Sun, "Symphony Generation with Permutation Invariant Language Model," *Conference of the International Society for Music Information Retrieval (ISMIR)*, 2022.
- [10] Y. Tay, M. Dehghani, D. Bahri, and D. Metzler, "Efficient transformers: A survey," *ACM Computing Surveys*, vol. 55, no. 6, 2022.
- [11] Y.-H. H. Tsai, S. Bai, M. Yamada, L.-P. Morency, and R. Salakhutdinov, "Transformer Dissection: An Unified Understanding for Transformer's Attention via the Lens of Kernel," *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.
- [12] M. Agarwal, C. Wang, and G. Richard, "F-stripe: Fast structure-informed positional encoding for symbolic music generation," *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025.
- [13] A. Liutkus, O. Cifka, S.-L. Wu, U. Simsekli, Y.-H. Yang, and G. Richard, "Relative Positional Encoding for Transformers with Linear Complexity," *International Conference on Machine Learning (ICML)*, 2021.
- [14] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2007.
- [15] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu, "Roformer: Enhanced transformer with rotary position embedding," *Neurocomputing*, vol. 568, p. 127063, 2024.
- [16] G. Ke, D. He, and T. Liu, "Rethinking positional encoding in language pre-training," *International Conference on Learning Representations (ICLR)*, 2021.
- [17] T. Chang, Y. Xu, W. Xu, and Z. Tu, "Convolutions and self-attention: Re-interpreting relative positions in pre-trained language models," *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2021.
- [18] P.-C. Chen, H. Tsai, S. Bhojanapalli, H. W. Chung, Y.-W. Chang, and C.-S. Ferng, "A simple and effective positional encoding for transformers," *Empirical Methods in Natural Language Processing (EMNLP)*, 2021.
- [19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention Is All You Need," *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2017.
- [20] K. M. Choromanski, V. Likhoshesterov, D. Dohan, X. Song, A. Kane, T. Sarlos, P. Hawkins, J. Q. Davis, A. Mohiuddin, L. Kaiser, D. B. Belanger, L. J. Colwell, and A. Weller, "Rethinking Attention with Performers," *International Conference on Machine Learning (ICML)*, 2021.
- [21] A. Katharopoulos, A. Vyas, N. Pappas, and F. Fleuret, "Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention," *International Conference on Machine Learning (ICML)*, 2020.
- [22] S. Zhuoran, Z. Mingyuan, Z. Haiyu, Y. Shuai, and L. Hongsheng, "Efficient Attention: Attention with Linear Complexities," *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021.
- [23] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," *Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2018.
- [24] S. Black, S. Biderman, E. Hallahan, Q. Anthony, L. Gao, L. Golding, H. He, C. Leahy, K. McDonell, J. Phang, M. Pieler, U. S. Prashanth, S. Purohit, L. Reynolds, J. Tow, B. Wang, and S. Weinbach, "GPT-NeoX-20B: An open-source autoregressive language model," *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2022.
- [25] G. Team, T. Mesnard, C. Hardin, R. Dadashi, S. Bhupatiraju, S. Pathak, L. Sifre, M. Riviere, M. S. Kale, J. Love, P. Tafti, L. Hussenot, P. G. Sessa, A. Chowdhery, A. Roberts, A. Barua, A. Botev, A. Castro-Ros, A. Slone, A. Hliou, A. Tacchetti, A. Bulanova, A. Paterson, B. Tsai, B. Shahriari, C. L. Lan, C. A. Choquette-Choo, C. Crepy, D. Cer, D. Ippolito, D. Reid, E. Buchatskaya, E. Ni, E. Noland, G. Yan, G. Tucker, G.-C. Muraru, G. Rozhdestvenskiy, H. Michalewski, I. Tenney, I. Grishchenko, J. Austin, J. Keeling, J. Labanowski, J.-B. Lespiau, J. Stanway, J. Brennan, J. Chen, J. Ferret, J. Chiu, J. Mao-Jones, K. Lee, K. Yu, K. Millican, L. L. Sjoesund, L. Lee, L. Dixon, M. Reid, M. Mikua, M. Wirth, M. Sharman, N. Chinaev, N. Thain, O. Bachem, O. Chang, O. Wahltinez, P. Bailey, P. Michel, P. Yotov, R. Chaabouni, R. Comanescu, R. Jana, R. Anil, R. McIlroy, R. Liu, R. Mullins, S. L. Smith, S. Borgeaud, S. Girgin, S. Douglas, S. Pandya, S. Shakeri, S. De, T. Klimenko, T. Hennigan, V. Feinberg, W. Stokowiec, Y. hui Chen, Z. Ahmed, Z. Gong, T. Warkentin, L. Peran, M. Giang, C. Farabet, O. Vinyals, J. Dean, K. Kavukcuoglu, D. Hassabis, Z. Ghahramani, D. Eck, J. Barral, F. Pereira, E. Collins, A. Joulin, N. Fiedel, E. Senter, A. Andreev, and K. Kenealy, "Gemma: Open models based on gemini research and technology," 2024. [Online]. Available: <https://arxiv.org/abs/2403.08295>
- [26] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozire, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "Llama: Open and efficient foundation language models," 2023. [Online]. Available: <https://arxiv.org/abs/2302.13971>
- [27] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel, "Palm: scaling language modeling with pathways," *J. Mach. Learn. Res.*, vol. 24, no. 1, Jan. 2023.
- [28] kaikonde, "Things i'm learning while training superhot," 2023. [Online]. Available: <https://kaikondet.github.io/tl>
- [29] S. Chen, S. Wong, L. Chen, and Y. Tian, "Extending context window of large language models via positional interpolation," *arXiv*, 2023. [Online]. Available: <https://arxiv.org/abs/2306.15595>
- [30] S. Wang, I. Kobyzev, P. Lu, M. Rezagholizadeh, and B. Liu, "Resonance rope: Improving context length generalization of large language models," *Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.
- [31] S. Pawar, S. M. T. I. Tonmoy, S. M. M. Zaman, V. Jain, A. Chadha, and A. Das, "The what, why, and how of context length extension techniques in large language models – a detailed survey," *arXiv*, 2024. [Online]. Available: <https://arxiv.org/abs/2401.07872>
- [32] F. Barbero, A. Vitvitskiy, C. Perivolaropoulos, R. Pascanu, and P. Velickovi, "Round and round we go! what makes rotary positional encodings useful?" *International Conference on Learning Representations (ICLR)*, 2025.
- [33] B. Scholkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2001.
- [34] A. Rahimi and B. Recht, "Weighted sums of random kitchen sinks: replacing minimization with randomization in learning," *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2008.
- [35] Z. Qin, W. Sun, K. Lu, H. Deng, D. Li, X. Han, Y. Dai, L. Kong, and Y. Zhong, "Linearized relative positional encoding," *Transactions on Machine Learning Research*, 2023.
- [36] Z. Wang, K. Chen, J. Jiang, Y. Zhang, M. Xu, S. Dai, X. Gu, and G. Xia, "POP909: A Pop-song Dataset for Music Arrangement Generation," *Conference of the International Society for Music Information Retrieval (ISMIR)*, 2020.

- [37] S. Dai, H. Zhang, and R. B. Dannenberg, "Automatic Analysis and Influence of Hierarchical Structure on Melody, Rhythm and Harmony in Popular Music," *Joint Conference on AI Music Creativity (AIMC)*, 2020.
- [38] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," *International Conference on Machine Learning (ICML)*, 2009.
- [39] J.-C. Wang, J. B. L. Smith, W.-T. Lu, and X. Song, "Supervised metric learning for music structure features," *Conference of the International Society for Music Information Retrieval (ISMIR)*, 2021.
- [40] S.-L. Wu and Y.-H. Yang, "Musemorphose: Full-song and fine-grained piano music style transfer with one transformer vae," *IEEE/ACM Transactions on Audio, Speech, & Language Processing (TASLP)*, vol. 31, pp. 1953–1967, 2023.
- [41] B. Haki, M. Nieto, T. Pelinski, and S. Jordà Puig, "Real-time drum accompaniment using transformer architecture," *Joint Conference on AI Music Creativity (AIMC)*, 2022.
- [42] D. S. Moore and S. Kirkland, *The Basic Practice of Statistics*, 2nd ed. W. H. Freeman & Co., 1999.
- [43] H. Levene, "Robust tests for equality of variances," *Contributions to probability and statistics. Essays in honor of Harold Hotelling*, pp. 279–292, 1961.
- [44] B. L. Welch, "The generalization of 'student's' problem when several different population variances are involved," *Biometrika*, vol. 34, no. 1/2, pp. 28–35, 1947.
- [45] A. Favero, A. Sclocchi, F. Cagnetta, P. Frossard, and M. Wyart, "How compositional generalization and creativity improve as diffusion models are trained," *arXiv*, 2025. [Online]. Available: <https://arxiv.org/abs/2502.12089>
- [46] K. Mei, H. Talebi, M. Ardakani, V. M. Patel, P. Milanfar, and M. Delbracio, "The power of context: How multimodality improves image super-resolution," *arXiv*, 2025. [Online]. Available: <https://arxiv.org/abs/2503.14503>