Learning a Fourier Transform for Linear Relative Positional Encodings in Transformers

Krzysztof Marcin Choromanski[*1,2] **Shanda Li**[*3] **Valerii Likhosherstov**[4]
Avinava Dubey[1] **Shengjie Luo**[5] **Di He**[5] **Yiming Yang**[3]
Tamas Sarlos[1] **Thomas Weingarten**[1] **Adrian Weller**[6,7]
[1]Google Research [2]Columbia University [3]Carnegie Mellon University
[4]Waymo [5]Peking University [6]University of Cambridge [7]Alan Turing Institute
{kchoro,avinavadubey}@google.com shandal@cs.cmu.edu

[*]These two authors contributed equally. The authorship is in alphabetical order.

Abstract

We propose a new class of linear Transformers called FourierLearner-Transformers (FLTs), which incorporate a wide range of relative positional encoding mechanisms (RPEs). These include regular RPE techniques applied for sequential data, as well as novel RPEs operating on geometric data embedded in higher-dimensional Euclidean spaces. FLTs construct the optimal RPE mechanism implicitly by learning its spectral representation. As opposed to other architectures combining efficient low-rank linear attention with RPEs, FLTs remain practical in terms of their memory usage and do not require additional assumptions about the structure of the RPE mask. Besides, FLTs allow for applying certain structural inductive bias techniques to specify masking strategies, e.g. they provide a way to learn the so-called *local RPEs* introduced in this paper and give accuracy gains as compared with several other linear Transformers for language modeling. We also thoroughly test FLTs on other data modalities and tasks, such as image classification, 3D molecular modeling, and learnable optimizers. To the best of our knowledge, for 3D molecular data, FLTs are the first Transformer architectures providing linear attention and incorporating RPE masking.

Proceedings of the 27[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238.

1 INTRODUCTION

Transformers have revolutionized the landscape of machine learning, introducing a paradigm shift in the way that people approach complex tasks in natural language processing (NLP) [Devlin et al., 2019], computer vision (CV) [Dosovitskiy et al., 2021], molecular modeling [Jumper et al., 2021], and beyond.

The largest computational bottleneck in Transformers is also the source of their success, the attention module. The attention module propagates signals between different tokens in the input sequence and has quadratic time and space complexity with respect to the input length L, which limits its scalability to long sequences. Thus, designing efficient attention modules has been an active area of research. Recently, the research on "efficient" Transformers has taken on new importance as the size of Transformer models grew from the GPT-1 architecture of "only" **117M** parameters to GPT-3 with **175B** parameters, a **1000**$\times$ increase within just two years [Brown et al., 2020].

One class of efficient Transformers is based on *sparse attention* [Li et al., 2019, Vaswani et al., 2021a, Zaheer et al., 2020, Roy et al., 2021, Vyas et al., 2020, Kitaev et al., 2020, Sun et al., 2022]. These methods do not aim at approximating the regular attention, but rather propose simpler and more tractable attention mechanisms, sometimes with additional constraints (e.g. identical queries and keys [Kitaev et al., 2020]). Another popular class of efficient Transformers is based on the *kernelized attention* [Choromanski et al., 2021, Tsai et al., 2019, Katharopoulos et al., 2020]. The key idea is to find an approximate low-rank decomposition of the attention matrix and leverage it to improve space and time complexity of the attention mechanism via the associativity property of matrix multiplications. Performer [Choromanski et al., 2021] is a

successful example of this model class. In contrast to previously discussed methods, Performer's approximate attention matrix (which is never explicitly constructed but rather implicitly used) is an unbiased estimate of the original attention matrix encoding similarities between tokens via the softmax kernel. Performers have been adopted into many Transformer stacks to provide linear space and time complexity [Yuan et al., 2021, Horn et al., 2021, Tay et al., 2021, Xiao et al., 2022].

Unfortunately, the simplicity of Performers comes at a price. It is well known that incorporating structural inductive priors which is usually implemented via various additive relative masking mechanisms in regular attention architectures is difficult for Performers. We refer to these methods as Relative Positional Encodings (RPEs) [Shaw et al., 2018, Raffel et al., 2020, Li et al., 2021, Luo et al., 2022]. RPEs play a critical role in improving the performance of Transformers in long-range modeling for language [Dai et al., 2019], speech [Liutkus et al., 2021], vision [Wu et al., 2021], and genomic data [Riga Avsec et al., 2021]. However, at first glance, Performers are not compatible with general RPE techniques, since they seem to require explicit materialization of the attention matrix to apply the RPE mask, which is exactly what Performers avoid in order to achieve computational improvements. Substantial efforts are made to reconcile Performers with RPEs (more details in Sec. 2), but so far all these attempts fall short of providing two properties at the same time: (a) practical computational gains, and (b) inclusion of general RPE methods, for inputs with nontrivial topological structures.

In this paper, we propose a new class of linear Transformers called FourierLearner-Transformers (FLTs), which incorporate a wide range of relative positional encoding mechanisms (RPEs). These include regular RPE techniques applied for sequential data, and novel RPEs operating on geometric data embedded in higher-dimensional Euclidean spaces (e.g. molecular structures). FLTs construct the optimal RPE mechanism implicitly by learning its spectral representation, and enjoy provable uniform convergence guarantees. As opposed to other architectures combining efficient low-rank linear attention with RPEs, FLTs remain practical in terms of their memory usage and do not require additional assumptions about the structure of the RPE mask. Besides, FLTs allow the application of certain structural inductive bias techniques to specify masking strategies, e.g. they provide a way to learn what we call local RPEs, introduced in this paper and providing accuracy gains compared with several other linear Transformers for language modeling. We also thoroughly test FLTs on other data modalities and tasks, such as image classification and molecular modeling. To the best of our knowledge, for 3D molecular data, FLTs are the first Transformer architectures providing linear attention and incorporating RPE masks, which broadens the scope of RPE-enhanced linear attention.

To summarize, our main contributions are as follows:

- We introduce the proposed RPE-enhanced linear attention, FourierLearner-Transformers (FLTs). FLTs are applicable to not only sequential data (e.g., texts) but also geometric data embedded in higher-dimensional Euclidean spaces (e.g., 3D molecular data), significantly broadening the scope of RPE-enhanced linear attention.
- We provided detailed theoretical analysis on FLTs, including the uniform convergence and sample complexity bound on its approximation (Sec. 4.1). We discuss several instantiations, in particular FLTs with so-called Gaussian mixture RPEs, shift-invariant kernel RPEs and local RPEs (Sec. 4.3).
- We extensively evaluate FLTs on language modeling (Sec. 5.1), image classification (Sec. 5.2), and molecular property predictions (Sec. 5.3). Our experiments show that FLTs can be easily applied to a wide range of data modalities and demonstrate strong performance and efficiency.

2 RELATED WORKS

Kernelized attention with RPE. One of the first attempts to address the problem of combining kernelized attention Transformers with RPEs is [Liutkus et al., 2021], where two variants, namely sineSPE and convSPE, are proposed. Both variants model the RPE mask as a stationary position kernel with a Toeplitz mask structure. While their complexity is linear in the sequence length L, extra dependency on the number of sinusoidal components T (for sineSPE) / the convolution filter lengths P (for convSPE) is introduced. In practice, T or P has to be sufficiently small due to computational budgets. Besides, they constrain the RPE mask to be a valid kernel matrix, while our FLTs do not require such assumptions. Both sineSPE and convSPE significantly underperform FLTs on language modeling (Sec. 5.1). And they cannot be applied for more general RPEs with tokens embedded in the higher-dimensional Euclidean spaces, e.g., RPEs for 3D molecular data.

Recently, [Luo et al., 2021, Choromanski et al., 2022a] show that the RPE mechanism can be combined with Performers in $O(L \log(L))$ time complexity. The method relies on the elegant observation that log-linear time complexity can be achieved as long as the exponentiated RPE mask supports fast matrix-vector

multiplication. RPEs for sequential data satisfy this condition since the corresponding masks have a Toeplitz structure. However, this method has large space complexity and high memory consumption in practice (Sec. 5.1). Moreover, it heavily relies on the structure of sequential data and does not apply to 3D molecular data where the RPE masks do not have a Toeplitz structure.

Random Fourier features (RFFs). There has been voluminous literature on the eld of RFFs [Rahimi and Recht, 2007, Avron et al., 2017, Szabó and Sriperumbudur, 2019, Li and Li, 2021, Choromanski et al., 2022b, Likhosherstov et al., 2022, Chowdhury et al., 2022]. However, the research on learnable RFF variants [Sinha and Duchi, 2016] is relatively new. Furthermore, prior works are mostly narrowed to applying RFFs in the context of positive de nite (PD) kernels, while our work breaks this limitation since RPEs do not need to be de ned by PD kernels. Several papers also explore the development of Transformer-based models whose attention mechanism operates in the spectral domain [Tamkin et al., 2020, Moreno-Pino et al., 2023], but they do not study e cient RPE modeling.

Long sequence modeling. Applying deep learning models to long sequences is an active research direction. We study e cient Transformer models for long sequence modeling. While our focus lies within the Transformer realm, it's worth noting the existence of alternative, non-Transformer architectures [Geng et al., 2021, Bello, 2021, Gu et al., 2022]. Beyond e ciency, [O'Connor and Andreas, 2021, Liu et al., 2024] probe context usage of long sequence language models; [Press et al., 2022, Ruoss et al., 2023, Li et al., 2024] design sequence models with length generalization ability (i.e., training on short sequences and generalize to long sequences); Yun et al. [2020], Yang et al. [2024] study the theoretical capability of those models. Note that existing works most focus on one data modality, while FLT is evaluated across a wide range modalities.

3 PRELIMINARIES

General RPE mechanism in Transformers. Consider an input sequence $X \in \mathbb{R}^{L \times d_{in}}$ where L and d_{in} denote the number and embedding size of tokens. The self-attention module in Transformers linearly projects the input into three matrices $Q, K, V \in \mathbb{R}^{L \times d}$ called queries, keys and values respectively. We also associate all the tokens with positional features $r_1, \ldots, r_L \in \mathbb{R}^{\ell}$ that are used to de ne the relative positional encoding (RPE) mechanism below:

De nition 3.1 (General RPE for attention). General Relative Positional Encoding enhanced attention is of the following form, where $N = [f(r_i - r_j)]_{i,j \in [L]} \in \mathbb{R}^{L \times L}$ is the so-called RPE mask[1] and $f : \mathbb{R}^{\ell} \to \mathbb{R}$ is a (potentially learnable) functon:

$$\mathrm{Att}(Q, K, V, N) = D^{-1}AV,$$
$$\text{where } A = \exp\left(N + \frac{QK^{\top}}{\sqrt{d}}\right), \; D = \mathrm{diag}(A 1_L). \quad (1)$$

Here $\exp(\cdot)$ is applied element-wise, 1_L is the all-one vector of length L, and $\mathrm{diag}(\cdot)$ is a diagonal matrix with the input vector as the diagonal. The time complexity of computing Eq. (1) is $O(L^2 d)$.

Discussions. De nition 3.1 is highly general because one can exibly choose the representation the positions r_i and the function f. For example, for sequential data like texts, positional indices in the sequence serves as the positional features ($r_i = i$), and the RPE mask is a learnable Topelitz matrix ($f(i - j) = c_{i-j}$ with parameters $\{c_k\}_{k=-(L-1)}^{L-1}$) [Ra el et al., 2020]. For geometric data like 3D molecular structures, one can view r_j as the 3D coordinates r_j of tokens (e.g., atoms) and use some domain-speci c f [Shi et al., 2022]. We emphasize that the general formulation is novel and important. It motivates the highly general FLTs applicable to a wide range of data and tasks, as opposed to existing approaches that heavily rely on the structure of sequential data and Toeplitz RPE masks (Sec. 2).

Kernelized linear attention. Kernelized attention techniques, e.g., Performers, leverage a decomposition of the attention matrix A to avoid explicit materialization of A, hence avoid the quadratic complexity in L. For the softmax attention, this is achieved by linearizing the softmax kernel $\exp(x^{\top} y)$ via random features, i.e., constructing for certain randomized mappings $\phi : \mathbb{R}^d \to \mathbb{R}^m$ such that $\exp(x^{\top} y) = \mathbb{E}[\phi(x)^{\top} \phi(y)]$.

De ne $Q', K' \in \mathbb{R}^{L \times m}$ as matrices of rows given as $\phi(q_i^{\top} d^{-\frac{1}{4}})^{\top}$ and $\phi(k_i^{\top} d^{-\frac{1}{4}})^{\top}$ respectively. Then the above linearization of softmax kernel directly leads to the following approximate algorithm for attention without RPE masks:

$$\widehat{\mathrm{Att}}_K(Q, K, V) = \widehat{D}^{-1}(Q'(K'^{\top}V))$$
$$\text{where } \widehat{D} = \mathrm{diag}(Q'(K'^{\top}1_L)). \quad (2)$$

Here $\widehat{\mathrm{Att}}_K$ stands for the approximate attention and brackets indicate the order of computations. The time and space complexity of this mechanism are $O(Lmd)$ and $O(Lm + md + Ld)$ respectively, compared to $O(L^2 d)$ and $O(L^2 + Ld)$ for regular attention. Thus, for $m \ll L$, Performers provide substantial computational improvements.

[1] We use $[L]$ to denote $\{1, \ldots, L\}$ in this paper.

4 METHOD

4.1 Efficient RPE-enhanced attention

The algorithm presented in Eq. (2) does not incorporate RPE mechanisms. In this subsection, we first present in Theorem 4.1 a novel technique to derive the (approximate) low rank decomposition of general RPE mask $\mathbf{N}$ in Definition 3.1. Next, we introduce FourierLearner-Transformer (FLT), a Performer-friendly RPE attention mechanism based on the decomposition.

Theorem 4.1. Given $f : \mathbb{R}^\ell \to \mathbb{R}$ and $\mathbf{N} = [f(\mathbf{r}_i - \mathbf{r}_j)] \in \mathbb{R}^{L \times L}$ as defined in Definition 3.1, denote by g the Fourier Transform of f. Assume p is some probability density function supported over $\mathbb{R}^\ell$. Sample $\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_r \overset{iid}{\sim} p$ and define the following random feature maps (where $i = \sqrt{-1}$):

$$\phi(\mathbf{z}) = \frac{1}{\sqrt{r}} e^{2\pi i \mathbf{z}^\top \boldsymbol{\omega}_1} \sqrt{\frac{g(\boldsymbol{\omega}_1)}{p(\boldsymbol{\omega}_1)}}, \ldots, e^{2\pi i \mathbf{z}^\top \boldsymbol{\omega}_r} \sqrt{\frac{g(\boldsymbol{\omega}_r)}{p(\boldsymbol{\omega}_r)}}\Big)^\top,$$

$$\psi(\mathbf{z}) = \frac{1}{\sqrt{r}} e^{-2\pi i \mathbf{z}^\top \boldsymbol{\omega}_1} \sqrt{\frac{g(\boldsymbol{\omega}_1)}{p(\boldsymbol{\omega}_1)}}, \ldots, e^{-2\pi i \mathbf{z}^\top \boldsymbol{\omega}_r} \sqrt{\frac{g(\boldsymbol{\omega}_r)}{p(\boldsymbol{\omega}_r)}}\Big)^\top,$$

Define $\mathbf{N}_1 = [\phi(\mathbf{r}_1), \ldots, \phi(\mathbf{r}_L)]^\top \in \mathbb{R}^{L \times r}$ and $\mathbf{N}_2 = [\psi(\mathbf{r}_1), \ldots, \psi(\mathbf{r}_L)]^\top \in \mathbb{R}^{L \times r}$. Then $\mathbb{E}[\mathbf{N}_1 \mathbf{N}_2^\top] = \mathbf{N}$.

Performer-friendly RPE attention. Theorem 4.1 implies that $\widehat{\mathbf{N}} = \mathbf{N}_1 \mathbf{N}_2^\top$ is a low-rank unbiased estimator of $\mathbf{N}$. Consequently, a Performer-friendly RPE attention mechanism with linear complexity can be obtained. Specifically, let $\widehat{\mathbf{Q}} = [\mathbf{N}_1, \mathbf{Q}' d^{\frac{1}{4}}] \in \mathbb{R}^{L \times (m+r)}$, $\widehat{\mathbf{K}} = [\mathbf{N}_2, \mathbf{K}' d^{\frac{1}{4}}] \in \mathbb{R}^{L \times (m+r)}$ where the concatenation is conducted along the second axis. Then

$$\widehat{\mathbf{A}} \overset{\text{def}}{=} \exp\Big(\widehat{\mathbf{N}} + \frac{\mathbf{Q}'\mathbf{K}'^\top}{\sqrt{d}}\Big) = \exp\big(\widehat{\mathbf{Q}}\widehat{\mathbf{K}}^\top\big) \quad (3)$$

In Eq. (3), RPE-masked attention is now translated to regular softmax attention that admits Performerization as described in Eq. (2). This observation naturally leads to an efficient RPE-enhanced attention algorithm, with a pseudo-code implementation provided in Algorithm 1. The time and space complexity of the algorithm are $O(L(m+r)d)$ and $O(L(m+r)+(m+r)d+Ld)$, respectively.

In Algorithm 1, instead of learning f and trying to compute its Fourier Transform g for the low-rank decomposition of $\mathbf{N}$, we propose to directly learn g and refer to our approach as FourierLearner-Transformer (FLT). Note that FLT effectively learns a spectral representation of f.

We point out that our formulations are general enough to cover a wide range of RPE variants used in practice:

Regular RPE for sequential data. In this setting the input sequence does not have richer geometric structure and thus vectors $\mathbf{r}_j$ can be identified as the indices of tokens in the sequence, i.e. $\mathbf{r}_j = j$. Thus, FLT learns a function $g : \mathbb{R} \to \mathbb{C}$ (Sec. 5.1, 5.2).

RPE for 3D-data. For this input type (e.g. 3D molecular data), it is natural to identify $\mathbf{r}_j$ as the 3D coordinates of atoms. Thus, FLT learns a function $g : \mathbb{R}^3 \to \mathbb{C}$. Note that existing methods [Liutkus et al., 2021, Luo et al., 2021, Choromanski et al., 2022a] are inapplicable while FLT works well in this setting (Sec. 5.3).

Finally, we note that FLT necessitates specifying some distribution p supported over $\mathbb{R}^\ell$ to satisfy the assumption in Theorem 4.1. Practical considerations dictate that p needs to be chosen in such a way that we can efficiently sample from it and compute its density function. In our experiments, we use Gaussian distributions zero mean and unit variance/learnable variance for p.

4.2 Theoretical analysis of FLT

We have theoretically investigated FLT's RPE approximation. In particular, we prove the following theorem, which states that under mild assumptions, the estimated RPE mask $\widehat{\mathbf{N}}$ can approximate the true RPE mask $\mathbf{N}$ up to arbitrary precision with high probability. Besides, the theorem provides sample complexity bound for such accurate approximation.

Theorem 4.2 (Uniform convergence and sample complexity for approximation). Given L vectors $\mathbf{r}_1, \ldots, \mathbf{r}_L \in \mathbb{R}^\ell$, define the RPE attention mask $\mathbf{N} = [f(\mathbf{r}_i - \mathbf{r}_j)]_{i,j \in [L]}$. Assume that $c = \|g(\mathbf{x})\|/p(\mathbf{x})\|_1$, where g is the Fourier Transform of f and p is some probability density function over $\mathbb{R}^\ell$.

For any $\varepsilon, \delta > 0$, if the number of random features $r = \frac{c^2}{\varepsilon^2} \log \frac{L}{\delta}$, then FLT's RPE approximator $\widehat{\mathbf{N}}$ satisfies

$$P\big(\|\mathbf{N} - \widehat{\mathbf{N}}\|_{\max} \leq \varepsilon\big) > 1 - \delta,$$

where $\|\cdot\|_{\max}$ denotes the max norm of a matrix.

We also prove variance bound of the estimated RPE and present the result in the supplementary material. The proofs and detailed discussions on the theoretical results can be found in the supplementary material as well.

4.3 The topology of the Fourier Transform

Nowhere in the analysis in Sec. 4.1 have we relied on any structural properties of f. In particular, the matrix $\mathbf{N}$ does not need to be a valid positive definite kernel-matrix or even symmetric. However, if needed,

Algorithm 1 FourierLearner Transformer: linear-complexity RPE-enhanced attention

Require: Input queries, keys, values $Q; K; V \in \mathbb{R}^{L \times d}$ and positions $R \in \mathbb{R}^{L \times \ell}$; random feature map for attention (see Sec. 3); Fourier Transform of the RPE function g (potentially parametrized by θ).
Output: Approximate RPE-enhanced attention (Definition 3.1)

```
# Apply random feature maps for RPE approximation.
# φ' and φ are defined in Theorem 4.1 and applied column-wise; g is called in φ' and φ.
N_1 ← φ'(R), N_2 ← φ(R)
# Concatenate along the second axis.
Q̂ ← [N_1; Q d^{1/4}], K̂ ← [N_2; K d^{1/4}]
# Apply random feature map φ.
Q' ← φ(Q̂), K' ← φ(K̂)
# Kernelized linear attention (Sec. 3). Brackets indicate the order of computations.
B_1 ← Q'(K'^⊤ V), B_2 ← Q'(K'^⊤ 1_L), O ← diag(B_2)^{-1} B_1.
return O
```

desired inductive bias can be incorporated into FLT via certain parameterization schemes used to train g, as we discuss in this subsection.

Gaussian mixture RPEs. One of the most general parameterizations of g that we have considered is the so-called Gaussian mixture variant:

$$g(\theta) = \sum_{t=1}^{T} w_t \exp\left(\frac{\theta - \mu_t \|^2}{2\sigma_t^2}\right).$$

Therefore, the FT g is parameterized by $(2+\ell)T$ numbers: $w_1, \ldots, w_T, \mu_1, \ldots, \mu_T \in \mathbb{R}$, $\sigma_1, \ldots, \sigma_T \in \mathbb{R}^{\ell}$. In the special case where $T = 1$, the FT becomes a renormalized Gaussian kernel and as such, defines as another Gaussian kernel.

Shift-invariant kernels for RPE masks. It is straightforward to apply the FLT mechanism for RPEs to make N a kernel-matrix of any shift-invariant kernel [Rahimi and Recht, 2007]. By Bochner's Theorem, for a shift-invariant kernel: $K : \mathbb{R}^{\ell} \times \mathbb{R}^{\ell} \to \mathbb{R}$, there exists a corresponding probabilistic distribution p_K and some positive constant $C > 0$, such that

$$K(x, y) = C \int_{\mathbb{R}^d} e^{\mathbf{i}(x-y)^{\top}\omega} p_K(\omega) d\omega.$$

Thus, to obtain an unbiased approximation of the RPE mask N given by the kernel matrix $[K(s_i, s_k)]_{i,k=1,\ldots,L}$ for the shift-invariant kernel K, it suffices to take $r_j = \frac{1}{2} s_j$, $g(\theta) = C p_K(\theta)$ for $j = 1, \ldots, r$. Even if a particular class of shift-invariant kernels has been chosen, FLT still provides a way to learn its specific instantiation through learning an appropriately parameterized g.

Local RPEs. Through the corresponding structured parameterized Fourier Transforms g, FLT is also capable of modeling various schemes where the RPE mechanism needs to be applied only locally and regular attention is to be used for tokens far enough from each other. We call such strategies *local RPEs*. Local RPEs can be derived for both sequential data and high-dimensional geometric data.

The most basic local RPE takes $r_j = j$ and, for an attention radius $\nu > 0$ and $C \in \mathbb{R}$, defines f as[2]

$$f_{\nu;C}(r) = C \cdot \mathbf{I}[|r| \leq \nu]. \tag{4}$$

Such an RPE mechanism would (de)amplify the regular attention score between tokens close to each other by a certain multiplicative amount and might play a similar role as local attention [Vaswani et al., 2021b]. It turns out that the FT for such a f has a particularly simple form:

$$g_{f_{\nu;C}}(\theta) = C \cdot \frac{\sin(2\nu\theta)}{\theta}.$$

Interestingly, RPEs from Eq. (4) can be easily generalized to a higher-dimensional local RPE. In this case, we consider the positional encoding function $f : \mathbb{R}^{\ell} \to \mathbb{R}$ of the following form:

$$f_{\nu;C}(r) = \prod_{j=1}^{\ell} C \cdot \mathbf{I}[|r^{(j)}| \leq \nu_j] \quad (r \in \mathbb{R}^{\ell}),$$

where $r^{(j)}$ denotes the j-th entry of r. The corresponding Fourier Transform g can be factorized as

$$g_{f_{\nu;C}}(\theta) = C \prod_{j=1}^{\ell} \frac{\sin(2\nu_j\theta_j)}{\theta_j}.$$

This result can be further generalized. Consider the function g of the following form:

$$g^{\nu_1,\ldots,\nu_{\ell}}_{k_1,\ldots,k_{\ell}}(\theta) = C \prod_{j=1}^{\ell} \frac{\sin^{k_j}(2\nu_j\theta_j)}{\theta_j}$$

[2] Note that instead of using one indicator function in Eq. (4), one can also apply a linear combination of many with learnable radii and a list of coefficients.

The inverse Fourier Transform of g can be written as

$$f(\mathbf{r}) = M \prod_{j=1}^{d} f_j^{v_j}(\mathbf{r}_j);$$

where M is a constant and each $f_j^{v_j}$ is (a) continuous, (b) symmetric, (c) with compact support of length depending on v_j, and (d) piece-wise a polynomial of order k_j 1. Such functions f are natural candidates for continuous local RPE mechanisms for tokens with positions embedded in $\mathbb{R}^{\ell}$ and any ℓ 1. Examples of local RPE variants for $\ell = 2$, supported via FLT, are presented in Fig. 4 in Appendix C.1.

The above theoretical results can be directly obtained via straightforward integration and a realization that the N-dim FT of a function: $h(x_1; \ ; x_N) \stackrel{\text{def}}{=} h_1(x_1)$ $h_N(x_N)$ can be represented as the product of 1D FTs of the individual components h_j.

Remark. We point out that all the three parametrization schemes above are parameter-e cient. In all our experiments, FLT introduced < 0.03M additional parameters for relative positional encoding. Note that the number of additional parameters does not increase with the input sequence length.

5 EXPERIMENTS

In this section, we provide experimental results on diverse tasks to demonstrate the e ectiveness of the FLT architecture. We rst study language modeling with sequential text data, which is a standard setting for e cient RPE-enhanced attention and enables thorough comparisons with existing baselines. Next, we consider the computer vision domain and test FLTs on several image classi cation datasets. Finally, to show that FLTs broaden the scope of RPE-enhanced e cient Transformers, we experiment on molecular property prediction with complicated RPE masks that existing e cient RPE-enhanced attention baselines cannot handle. The complete experimental setup, the hyperparameters for each of the tasks, and hardware details are provided in the supplementary material.

5.1 Language modeling

We conduct experiments on the WikiText-103 language modeling task to show the e ectiveness of our proposed method in NLP applications. Most existing baselines are applicale to sequential text data. Thus, we provide comprehensive empirical comparisons on model quality and e ciency with baselines in this subsection.

Compared methods. In this experiment, we study FLT with two RPE vaiants, Gaussian mixture RPE and local RPE. We compare our model with the following strong baselines:

Table 1: Language model perplexity scores on the WikiText-103 validation set. The lowest perplexity is highlighted in bold.

Model	Perplexity
Linear Trans. [Katharopoulos et al., 2020]	38.4
RFA-Gaussian [Peng et al., 2021]	33.6
RFA-arccos [Peng et al., 2021]	36.0
RFA-GATE-Gaussian [Peng et al., 2021]	31.3
RFA-GATE-arccos [Peng et al., 2021]	32.8
Performer [Choromanski et al., 2021]	31.1
CosFormer [Qin et al., 2022]	30.7
Performer-sineSPE [Liutkus et al., 2021]	38.0
Performer-convSPE [Liutkus et al., 2021]	37.8
Log-linear Performer [Luo et al., 2021]	30.6
FLT (Gaussian mixture RPE) (ours)	30.3
FLT (local RPE) (ours)	30.1

- Linear Transformer [Katharopoulos et al., 2020], which uses kernelized low-rank attention with elu( ) + 1 as the feature map.
- Random feature attention (RFA) [Peng et al., 2021], which has two variants (Gaussian and arc-cosine) and an optional gating mechanism.
- The regular Performer [Choromanski et al., 2021], which applies the FAVOR+ mechanism for attention matrix approximation.
- CosFormer [Qin et al., 2022], which designs a linear operator and a cosine-based distance reweighting mechanism for attention matrix approximation.
- Performer-SPE [Liutkus et al., 2021], which incorporates a special class of RPE into low-rank attention and has two variants (sineSPE and convSPE).
- The log-linear Performer [Luo et al., 2021] which extends Performers to work with an arbitrary Toeplitz RPE attention mask.

Implementation details. All the tested models are e cient Transformers based on kernelized low-rank attention, with 6 decoder layers. More details regarding model con gurations and training are in the supplementary material. We use the validation perplexity as the evaluation metric; lower perplexity indicates better performance.

Figure 1: Model forward speed (left) and peak memory (right) comparisons between FLT and baselines under different input sequence lengths.

Results. The results are shown in Table 1. Both variants of our FLT outperform all the baselines. Compared with efficient Transformers without RPE, FLT achieves much stronger performance. For example, the validation perplexity of our FLT with local RPE is 1.0 point lower than that of the regular Performer, indicating that our method brings substantial performance gains by incorporating RPE into the attention module.

Compared with other efficient Transformer variants with RPE, our FLT is still very competitive. For example, our FLT achieves lower perplexity than the strong log-linear Performer baseline. Note that log-linear Performer relies on more expensive FFT and is less efficient in practice. Specifically, the time and space complexity of the FLT are $O(L(m+r)d)$ and $O(L(m+r)+(m+r)d+Ld)$, respectively, while the time and space complexity of log-linear Performer are $O(Lmd\log L)$ and $O(Lmd)$. Thus, our FLT obtains both better quality and efficiency than existing efficient RPE-enhanced Transformer variants on this task.

In addition, we further investigate the attention matrices of FLT [3]. We visualize the attention matrices of different attention heads in an FLT model trained on WikiText-103 language modeling in Fig. 5 in Appendix C.2. The visualizations show that some attention heads pay more attention to nearby tokens, while others shows global attention patterns. Quantitatively, the average attention probability over the most distant/nearby 10% tokens is 0.068/0.279 respectively. Thus, FLT learns locality bias in language while maintaining the advantage to capture global contexts and leverage information in distant tokens.

Computational cost comparisons. As discussed above, FLT enjoys much better time/space complexity compared with the strongest baseline method, the log-linear Performer. To showcase FLT 's efficiency in practice, we construct one Transformer layer with 12 attention heads whose hidden dimension is set to 768, and FFN dimension is set to 3072. We feed inputs with varying lengths and a batch size of 8 into the model and measure the efficiency. We report the average forward time and the maximum peak memory consumption across 5 runs under different input sequence lengths in Fig. 1. We compare FLT with the strongest baseline, log-linear Performer, and we also include the regular Performer as a reference. It's clear that FLT only introduces negligible memory overhead compared with the regular Performer, and scales much better than the log-linear Performer in practice, in terms of both model forward time and peak memory. Therefore, our experiment show that FLT is both more accurate and more scalable than the baselines on sequential text data.

[3] FLT does not explicitly construct attention matrices during training so that it avoids the quadratic computational complexity. However, we can still materialize the attention matrices approximated by FLT

Table 2: Image classification accuracy comparisons. Log-linear Performer is omitted due to its infeasible memory complexity and out-of-memory issues. The best performances are highlighted in bold.

	ImageNet	Places365	FashionMnist
Performer	75.1%	55.0%	91.1%
CosFormer	76.2%	55.6%	91.6%
FLT (ours)	**77.4%**	**56.0%**	**92.1%**

5.2 Image classification

We thoroughly benchmarked FLT variants of Vision Transformers (ViTs) [Dosovitskiy et al., 2021] on several image classification datasets, including ImageNet, Places365, and FashionMnist. Details of these datasets can be found in the supplementary material.

Compared methods and implementation details. We compare FLT with the regular Performer as well as the most competitive competitor from Sec. 5.1, CosFormer and log-linear Performer. All tested ViTs consist of 12 layers with 12 attention heads in each layer. More details regarding model configurations and training are in the supplementary material. For

Table 3: Comparisons of FLT with the regular Performer on OC20 IS2RE task. The su x -kL means the model consists ofk layers, e.g.,FLT -10L refers to a 10-layer FLT . The evaluation metrics are Mean Absolute Error (MAE, lower is better) of the energies and the percentage of Energies within a Threshold (EwT, higher is better). We highlighted in bold the best performance.

	Energy MAE (eV)	EwT (%)
Performer-12L	0.5454	4.90
FLT -10L (ours)	0.5157	5.44
FLT -12L (ours)	**0.5046**	**5.33**

Figure 2: Validation loss of FLTs and the regular Performer on the IS2RE task of OC20 dataset.

our FLT variants, we apply Gaussian mixture RPEs (Sec. 4.3) with the number of Gaussian mixture modes T set to 25 and the number of random features for RPE-encoding r set to 64.

Results. The results are presented in Table 2. The log-linear Performer architecture run out of memory for m = 128 and does not train when m was reduced (with a xed batch size of 4096) to t the assigned memory. Thus, it is omitted in the comparison. Compared with the other two baselines, our FLT obtains strongest performances on all the three datasets. For instance, on ImageNet,FLT provides a 2:3% accuracy improvement over the regular Performer; and is even 1:2% better than the strong CosFormer baseline. The results demonstrate that FLT also works well on image data.

5.3 Molecular property prediction

As highlighted in previous discussions,FLT broadens the scope of RPE-enhanced e cient Transformers and can be applied to geometric data embedded in high-dimensional Euclidean spaces. To validate this claim, in this subsection, we further evaluate our FLT model on the molecular property prediction task to show its capability to handle 3D input data and complicated (non-Toeplitz) RPE masks. To the best of our knowledge, in this scenario,FLT is the rst Transformer providing RPE-enhanced scalable attention that enjoys linear complexity with respect to the number of input tokens.

We use a publicly-available large-scale electrocatalysts dataset - the Open Catalyst 2020 (OC20) dataset and focus on the IS2RE task which requires to predict the energy of the relaxed structure given the initial structure of solid catalysts with adsorbate molecules [Chanussot* et al., 2021].

Compared methods. Existing technicques considered in the previous experiments do not apply to this setting. Thus, we only compare our FLT with the regular Performer without RPE. For the FLT model, we consider to approximate RPE masks based on Gaussian basis functions, which are popularly used in neural networks for molecular modeling [Gasteiger et al., 2021; Shi et al., 2022; Luo et al., 2023]. Speci cally, the RPE mask is de ned as $N = [f(r_i - r_j)]_{i,j \in [L]} \in \mathbb{R}^{L \times L}$, where $r_i \in \mathbb{R}^3$ is the position of the i-th input atom, L is the total number of input atom, and

$$f(r) = \sum_{t=1}^{T} \frac{w_t}{(\sqrt{2\pi}\sigma_t)^3} \exp\left(\frac{\|r\|^2}{2\sigma_t^2}\right);$$

Note that RPE only calculates the relative distances between atoms, which naturally preserves many invariant and equivariant properties. It easy to see that the Fourier Transform of f is

$$g(\omega) = \sum_{t=1}^{T} w_t \exp\left(-2\pi^2\sigma_t^2\|\omega\|^2\right);$$

which enables us to approximate the RPE maskN in FLTs using the technique described in Sec. 4.

Implementation details. We adopt most of the training strategies of 3D-Graphormer [Shi et al., 2022]. Speci cally, we trained a regular Performer with 12 layers and two FLT with 10 and 12 layers respectively. More details regarding model con gurations and training are in the supplementary material. We evaluate the model performance on the in-domain validation set. We use Mean Absolute Error (MAE) of the energies and the percentage of Energies within a Threshold (EwT) of the ground truth energy to evaluate the accuracy of the predicted energies.

Results. The results are presented in Table 3. We also present the validation loss curves of the models in Fig. 2 for a more comprehensive comparison. Clearly, our FLT models obtain better performance in both evaluation metrics and produce more accurate energy predictions. For example, the energy MAE of the 12-layer FLT is more than 0.04eV lower than that of the

Figure 3: Results of learnable optimizer experiments. Left: Adam & learnable optimizers using FLT and S4 on the task of training ViT-Base classifier on ImageNet. Right: Adam & various learnable optimizers on the task of optimizing Rastrigin-type functions (from private conversation with the authors of [Jain et al., 2023]).

12-layer regular Performer, which indicates that the use of RPE effectively increases the predictive power of the model. One may argue that the use of RPE in FLT may add some computational overhead and increase the number of model parameters. However, it should be noted that a shallower 10-layer FLT can also significantly outperform the 12-layer regular Performer, while being faster and using less parameters.

5.4 Learnable optimizers

FLT has also been compared independently by authors and other researchers on longer contexts with other classes of efficient architecture, including LSTM [Hochreiter and Schmidhuber, 1997] and state-space models [Gu et al., 2022]. The corresponding task is practical and challenging: applying Transformers as memory models in learnable optimizers (with context length up to 2000). In this setting, long-range temporal (to understand the history of the optimization) and spatial (to understand the landscape of the loss function more globally) reasoning is critical [Jain et al., 2023, Gärtner et al., 2023].

Compared methods and implementation details. In the first experiment, FLT-based learnable optimizer is compared against S4-based learnable optimizer [Gu et al., 2022] and standard non-learnable Adam optimizer on ImageNet classification.

In the second experiment, FLT is further applied on population(swarm)-based learnable optimizers, in which the masking mechanism implemented by FLT was applied to modulate how the members of the population attend to each other. The baselines include standard non-learnable Adam optimizer as well as learnable optimizers based on LSTM and Performer. The evaluation is conducted on Rastrigin-like functions.

Results. The results of the first/second experiment are shown in the left/right panel of Fig. 3. We note that in both experiments, FLT-based approach provides drastic improvements over all other variants. The results shows the consistent effectiveness of our model in capturing long range dependencies in learnable optimizers.

6 CONCLUSIONS

We introduce FourierLearner-Transformers (FLTs) that efficiently adapt the relative positional encoding (RPE) mechanism into Performers - kernelized implicit-attention Transformers with linear space and time complexity. In contrast to other architectures combining Performers with RPEs, FLTs maintain linear complexity of the attention modules with no additional structural assumptions regarding the RPE mask. We provide theoretical analysis and show that FLTs can accurately approximate RPE. We further conduct extensive experiments to show the efficiency and quality of FLTs across a wide range of tasks and data modalities, including texts, images, molecules, and optimizer memory.

ACKNOWLEDGEMENTS

We would like to thank Deepali Jain for a discussion on using FLT for learnable optimizers, as well as proposing and performing experiments with population(swarm)-based methods for the FLT variants provided by the authors. We also thank the reviewers for their helpful comments.

This work is supported in part by the United States Department of Energy via the Brookhaven National Laboratory under Contract No. 384608.

References

[Avron et al., 2017] Avron, H., Kapralov, M., Musco, C., Musco, C., Velingker, A., and Zandieh, A. (2017). Random fourier features for kernel ridge regression: Approximation bounds and statistical guarantees. In Precup, D. and Teh, Y. W., editors, Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017, volume 70 of Proceedings of Machine Learning Research, pages 253–262. PMLR.

[Bello, 2021] Bello, I. (2021). Lambdanetworks: Modeling long-range interactions without attention. In International Conference on Learning Representations.

[Brown et al., 2020] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual

[Chanussot* et al., 2021] Chanussot*, L., Das*, A., Goyal*, S., Lavril*, T., Shuaibi*, M., Riviere, M., Tran, K., Heras-Domingo, J., Ho, C., Hu, W., Palizhati, A., Sriram, A., Wood, B., Yoon, J., Parikh, D., Zitnick, C. L., and Ulissi, Z. (2021). Open catalyst 2020 (oc20) dataset and community challenges. ACS Catalysis.

[Choromanski et al., 2022a] Choromanski, K., Lin, H., Chen, H., Zhang, T., Sehanobish, A., Likhosherstov, V., Parker-Holder, J., Sarlós, T., Weller, A., and Weingarten, T. (2022a). From block-Toeplitz matrices to differential equations on graphs: towards a general theory for scalable masked transformers. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S., editors, International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA, volume 162 of Proceedings of Machine Learning Research, pages 3962–3983. PMLR.

[Choromanski et al., 2021] Choromanski, K. M., Likhosherstov, V., Dohan, D., Song, X., Gane, A., Sarlós, T., Hawkins, P., Davis, J. Q., Mohiuddin, A., Kaiser, L., Belanger, D. B., Colwell, L. J., and Weller, A. (2021). Rethinking attention with performers. In 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net.

[Choromanski et al., 2022b] Choromanski, K. M., Lin, H., Chen, H., Sehanobish, A., Ma, Y., Jain, D., Varley, J., Zeng, A., Ryoo, M. S., Likhosherstov, V., Kalashnikov, D., Sindhwani, V., and Weller, A. (2022b). Hybrid random features. In The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022. OpenReview.net.

[Chowdhury et al., 2022] Chowdhury, S. P., Solomou, A., Dubey, A., and Sachan, M. (2022). On learning the transformer kernel. Transactions of Machine Learning Research

[Dai et al., 2019] Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q., and Salakhutdinov, R. (2019). Transformer-XL: Attentive language models beyond a fixed-length context. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 2978–2988, Florence, Italy. Association for Computational Linguistics.

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA, pages 248–255. IEEE Computer Society.

[Devlin et al., 2019] Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2019). BERT: pre-training of deep bidirectional transformers for language understanding. In Burstein, J., Doran, C., and Solorio, T., editors, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers), pages 4171–4186. Association for Computational Linguistics.

[Dosovitskiy et al., 2021] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. In 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net.

[Gärtner et al., 2023] Gärtner, E., Metz, L., Andriluka, M., Freeman, C. D., and Sminchisescu, C. (2023).

Transformer-based learned optimization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 11970 11979.

[Gasteiger et al., 2021] Gasteiger, J., Becker, F., and Günnemann, S. (2021). Gemnet: Universal directional graph neural networks for molecules. Advances in Neural Information Processing Systems, 34:6790 6802.

[Geng et al., 2021] Geng, Z., Guo, M.-H., Chen, H., Li, X., Wei, K., and Lin, Z. (2021). Is attention better than matrix decomposition? In International Conference on Learning Representations

[Gu et al., 2022] Gu, A., Goel, K., and Ré, C. (2022). E ciently modeling long sequences with structured state spaces. In The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022. OpenReview.net.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8):1735 1780.

[Horn et al., 2021] Horn, M., Shridhar, K., Groenewald, E., and Baumann, P. F. M. (2021). Translational equivariance in kernelizable attention. CoRR, abs/2102.07680.

[Jain et al., 2023] Jain, D., Choromanski, K. M., Dubey, K. A., Singh, S., Sindhwani, V., Zhang, T., and Tan, J. (2023). Mnemosyne: Learning to train transformers with transformers. In Thirty-seventh Conference on Neural Information Processing Systems.

[Jumper et al., 2021] Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., šídek, A., Potapenko, A., et al. (2021). Highly accurate protein structure prediction with alphafold. Nature, 596(7873):583 589.

[Katharopoulos et al., 2020] Katharopoulos, A., Vyas, A., Pappas, N., and Fleuret, F. (2020). Transformers are rnns: Fast autoregressive transformers with linear attention. In International Conference on Machine Learning, pages 5156 5165. PMLR.

[Kitaev et al., 2020] Kitaev, N., Kaiser, L., and Levskaya, A. (2020). Reformer: The e cient transformer. In 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020 OpenReview.net.

[Li et al., 2021] Li, S., Chen, X., He, D., and Hsieh, C.-J. (2021). Can vision transformers perform convolution? arXiv preprint arXiv:2111.01353.

[Li et al., 2019] Li, S., Jin, X., Xuan, Y., Zhou, X., Chen, W., Wang, Y.-X., and Yan, X. (2019). Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. Advances in neural information processing systems, 32.

[Li et al., 2024] Li, S., You, C., Guruganesh, G., Ainslie, J., Ontanon, S., Zaheer, M., Sanghai, S., Yang, Y., Kumar, S., and Bhojanapalli, S. (2024). Functional interpolation for relative positions improves long context transformers. In The Twelfth International Conference on Learning Representations.

[Li and Li, 2021] Li, X. and Li, P. (2021). Quantization algorithms for random fourier features. In Meila, M. and Zhang, T., editors, Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event, volume 139 of Proceedings of Machine Learning Research, pages 6369 6380. PMLR.

[Likhosherstov et al., 2022] Likhosherstov, V., Choromanski, K. M., Dubey, K. A., Liu, F., Sarlos, T., and Weller, A. (2022). Chefs' random tables: Non-trigonometric random features. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, Advances in Neural Information Processing Systems, volume 35, pages 34559 34573. Curran Associates, Inc.

[Liu et al., 2024] Liu, N. F., Lin, K., Hewitt, J., Paranjape, A., Bevilacqua, M., Petroni, F., and Liang, P. (2024). Lost in the middle: How language models use long contexts. Transactions of the Association for Computational Linguistics, 12:157 173.

[Liutkus et al., 2021] Liutkus, A., Cífka, O., Wu, S., Simsekli, U., Yang, Y., and Richard, G. (2021). Relative positional encoding for transformers with linear complexity. In Meila, M. and Zhang, T., editors, Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event , volume 139 of Proceedings of Machine Learning Research, pages 7067 7079. PMLR.

[Luo et al., 2023] Luo, S., Chen, T., Xu, Y., Zheng, S., Liu, T.-Y., Wang, L., and He, D. (2023). One transformer can understand both 2d & 3d molecular data. In The Eleventh International Conference on Learning Representations

[Luo et al., 2021] Luo, S., Li, S., Cai, T., He, D., Peng, D., Zheng, S., Ke, G., Wang, L., and Liu, T.-Y. (2021). Stable, fast and accurate: Kernelized attention with relative positional encoding. Advances in Neural Information Processing Systems, 34:22795 22807.

[Luo et al., 2022] Luo, S., Li, S., Zheng, S., Liu, T.-Y., Wang, L., and He, D. (2022). Your transformer may not be as powerful as you expect. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K., editors, Advances in Neural Information Processing Systems.

[Moreno-Pino et al., 2023] Moreno-Pino, F., Olmos, P. M., and Artés-Rodríguez, A. (2023). Deep autoregressive models with spectral attention. Pattern Recognition, 133:109014.

[O'Connor and Andreas, 2021] O'Connor, J. and Andreas, J. (2021). What context features can transformer language models use? In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 851 864.

[Peng et al., 2021] Peng, H., Pappas, N., Yogatama, D., Schwartz, R., Smith, N., and Kong, L. (2021). Random feature attention. In International Conference on Learning Representations

[Press et al., 2022] Press, O., Smith, N., and Lewis, M. (2022). Train short, test long: Attention with linear biases enables input length extrapolation. In International Conference on Learning Representations

[Qin et al., 2022] Qin, Z., Sun, W., Deng, H., Li, D., Wei, Y., Lv, B., Yan, J., Kong, L., and Zhong, Y. (2022). cosformer: Rethinking softmax in attention. CoRR, abs/2202.08791.

[Ra el et al., 2020] Ra el, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2020). Exploring the limits of transfer learning with a uni ed text-to-text transformer. J. Mach. Learn. Res., 21:140:1 140:67.

[Rahimi and Recht, 2007] Rahimi, A. and Recht, B. (2007). Random features for large-scale kernel machines. In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T., editors, Advances in Neural Information Processing Systems 20, Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 3-6, 2007, pages 1177 1184. Curran Associates, Inc.

[Roy et al., 2021] Roy, A., Sa ar, M., Vaswani, A., and Grangier, D. (2021). E cient content-based sparse attention with routing transformers. Trans. Assoc. Comput. Linguistics, 9:53 68.

[Ruoss et al., 2023] Ruoss, A., Delétang, G., Genewein, T., Grau-Moya, J., Csordás, R., Bennani, M., Legg, S., and Veness, J. (2023). Randomized positional encodings boost length generalization of transformers. arXiv preprint arXiv:2305.16843.

[Shaw et al., 2018] Shaw, P., Uszkoreit, J., and Vaswani, A. (2018). Self-attention with relative position representations. In Walker, M. A., Ji, H., and Stent, A., editors, Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers), pages 464 468. Association for Computational Linguistics.

[Shi et al., 2022] Shi, Y., Zheng, S., Ke, G., Shen, Y., You, J., He, J., Luo, S., Liu, C., He, D., and Liu, T.-Y. (2022). Benchmarking graphormer on large-scale molecular modeling datasets. arXiv preprint arXiv:2203.04810.

[Sinha and Duchi, 2016] Sinha, A. and Duchi, J. C. (2016). Learning kernels with random features. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R., editors, Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain, pages 1298 1306.

[Sun et al., 2022] Sun, Z., Yang, Y., and Yoo, S. (2022). Sparse attention with learning to hash. In The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022 . OpenReview.net.

[Szabó and Sriperumbudur, 2019] Szabó, Z. and Sriperumbudur, B. K. (2019). On kernel derivative approximation with random fourier features. In Chaudhuri, K. and Sugiyama, M., editors, The 22nd International Conference on Arti cial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan, volume 89 of Proceedings of Machine Learning Research, pages 827 836. PMLR.

[Tamkin et al., 2020] Tamkin, A., Jurafsky, D., and Goodman, N. (2020). Language through a prism: A spectral approach for multiscale language representations. Advances in Neural Information Processing Systems, 33:5492 5504.

[Tay et al., 2021] Tay, Y., Dehghani, M., Aribandi, V., Gupta, J., Pham, P. M., Qin, Z., Bahri, D., Juan, D.-C., and Metzler, D. (2021). Omninet: Omnidirectional representations from transformers. In International Conference on Machine Learning pages 10193 10202. PMLR.

[Tsai et al., 2019] Tsai, Y.-H. H., Bai, S., Yamada, M., Morency, L.-P., and Salakhutdinov, R. (2019). Transformer dissection: An uni ed understanding for transformer's attention via the lens of kernel. In

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) , pages 4344 4353.

[Vaswani et al., 2021a] Vaswani, A., Ramachandran, P., Srinivas, A., Parmar, N., Hechtman, B. A., and Shlens, J. (2021a). Scaling local self-attention for parameter e cient visual backbones. In IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021, pages 12894 12904. Computer Vision Foundation / IEEE.

[Vaswani et al., 2021b] Vaswani, A., Ramachandran, P., Srinivas, A., Parmar, N., Hechtman, B. A., and Shlens, J. (2021b). Scaling local self-attention for parameter e cient visual backbones. In IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021, pages 12894 12904. Computer Vision Foundation / IEEE.

[Vyas et al., 2020] Vyas, A., Katharopoulos, A., and Fleuret, F. (2020). Fast transformers with clustered attention. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual .

[Wu et al., 2021] Wu, K., Peng, H., Chen, M., Fu, J., and Chao, H. (2021). Rethinking and improving relative position encoding for vision transformer. In 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021, pages 10013 10021. IEEE.

[Xiao et al., 2017] Xiao, H., Rasul, K., and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. CoRR, abs/1708.07747.

[Xiao et al., 2022] Xiao, X., Zhang, T., Choromanski, K., Lee, T. E., Francis, A. G., Varley, J., Tu, S., Singh, S., Xu, P., Xia, F., Persson, S. M., Kalashnikov, D., Takayama, L., Frostig, R., Tan, J., Parada, C., and Sindhwani, V. (2022). Learning model predictive controllers with real-time attention for real-world navigation. CoRL 2022, abs/2209.10780.

[Yang et al., 2024] Yang, K., Ackermann, J., He, Z., Feng, G., Zhang, B., Feng, Y., Ye, Q., He, D., and Wang, L. (2024). Do e cient transformers really save computation? arXiv preprint arXiv:2402.13934 .

[Yuan et al., 2021] Yuan, L., Chen, Y., Wang, T., Yu, W., Shi, Y., Jiang, Z., Tay, F. E. H., Feng, J., and Yan, S. (2021). Tokens-to-token vit: Training vision transformers from scratch on imagenet. In 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021, pages 538 547. IEEE.

[Yun et al., 2020] Yun, C., Chang, Y.-W., Bhojanapalli, S., Rawat, A. S., Reddi, S., and Kumar, S. (2020). O(n) connections are expressive enough: Universal approximability of sparse transformers. Advances in Neural Information Processing Systems, 33:13783 13794.

[Zaheer et al., 2020] Zaheer, M., Guruganesh, G., Dubey, K. A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., et al. (2020). Big bird: Transformers for longer sequences. Advances in Neural Information Processing Systems, 33:17283 17297.

[Zhou et al., 2018] Zhou, B., Lapedriza, À., Khosla, A., Oliva, A., and Torralba, A. (2018). Places: A 10 million image database for scene recognition IEEE Trans. Pattern Anal. Mach. Intell. , 40(6):1452 1464.

[šiga Avsec et al., 2021] šiga Avsec, Agarwal, V., Visentin, D., Ledsam, J. R., Grabska-Barwinska, A., Taylor, K. R., Assael, Y., Jumper, J. M., Kohli, P., and Kelley, D. R. (2021). E ective gene expression prediction from sequence by integrating long-range interactions. Nature Methods, 18:1196 1203.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with speci cation of all dependencies, including external libraries. [No. The paper contains all the details (including pseudo code in Algorithm 1) needed to reproduce all the experiments conducted in the paper. All used datasets are publicly available.]
2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]
3. For all gures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes. The paper contains all the details (including pseudo code in Algorithm 1) needed to reproduce all the experiments conducted in the paper. All used datasets are publicly available.]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear de nition of the speci c measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identi able information or offensive content. [Not Applicable]
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A OMITTED THEORETICAL RESULTS AND PROOFS

We provide here omitted theoretical results, proofs, and discussions on FLT's RPE approximation. We first prove the RPE approximation proposed in the main paper is unbiased. Then we provide a variance bound and an approximation guarantee for it.

For convenience of reading, we always state the theorem before providing the proof, even if the theorem has appeared in the main body of the paper.

A.1 Unbiased RPE approximation

Theorem A.1. [4] *Given $f : \mathsf{R}^\ell \to \mathsf{R}$ and $\mathsf{N} = [f(\mathsf{r}_i - \mathsf{r}_j)] \in \mathsf{R}^{L\times L}$ as defined in Definition 3.1, denote by g the Fourier Transform of f. Assume p is some probability density function supported over R^ℓ. Sample $\xi_1, \cdots, \xi_r \overset{\text{iid}}{\sim} p$ and define the following random feature maps (where $\mathsf{i} = \sqrt{-1}$):*

$$\varphi(\mathsf{z}) = \frac{1}{\sqrt{r}} \Big(e^{2\pi \mathsf{i} \mathsf{z}^\top \mathsf{r}_1} \sqrt{\frac{g(\mathsf{r}_1)}{p(\mathsf{r}_1)}}, \cdots, e^{2\pi \mathsf{i} \mathsf{z}^\top \mathsf{r}_r} \sqrt{\frac{g(\mathsf{r}_r)}{p(\mathsf{r}_r)}} \Big)^\top;$$
$$\psi(\mathsf{z}) = \frac{1}{\sqrt{r}} \Big(e^{-2\pi \mathsf{i} \mathsf{z}^\top \mathsf{r}_1} \sqrt{\frac{g(\mathsf{r}_1)}{p(\mathsf{r}_1)}}, \cdots, e^{-2\pi \mathsf{i} \mathsf{z}^\top \mathsf{r}_r} \sqrt{\frac{g(\mathsf{r}_r)}{p(\mathsf{r}_r)}} \Big)^\top,$$

Define $\mathsf{N}_1 = [\varphi(\mathsf{r}_1), \cdots, \varphi(\mathsf{r}_L)]^\top \in \mathsf{R}^{L\times r}$ and $\mathsf{N}_2 = [\psi(\mathsf{r}_1), \cdots, \psi(\mathsf{r}_L)]^\top \in \mathsf{R}^{L\times r}$. Then

$$\mathsf{E}[\mathsf{N}_1 \mathsf{N}_2] = \mathsf{N}.$$

Proof. Be definition of N, it suffices to show that

$$f(\mathsf{r}_i - \mathsf{r}_j) = \mathsf{E}\ \varphi(\mathsf{r}_i)^\top \psi(\mathsf{r}_j)\ .$$

Note that g is the Fourier Transform of f. Therefore,

$$f(\mathsf{x}) = \int_{\mathsf{R}^d} e^{2\pi \mathsf{i} \mathsf{x}^\top \omega} g(\omega) \mathrm{d}\omega = \int_{\mathsf{R}^d} e^{2\pi \mathsf{i} \mathsf{x}^\top \omega} \frac{g(\omega)}{p(\omega)} \cdot p(\omega) \mathrm{d}\omega = \mathsf{E}_{\omega\sim p}\ e^{2\pi \mathsf{i} \mathsf{x}^\top \omega} \frac{g(\omega)}{p(\omega)}\ . \tag{5}$$
$$\Rightarrow \quad f(\mathsf{r}_i - \mathsf{r}_j) = \mathsf{E}_{\omega\sim p}\ e^{2\pi \mathsf{i} \mathsf{r}_i^\top \omega} \sqrt{\frac{g(\omega)}{p(\omega)}} \cdot e^{-2\pi \mathsf{i} \mathsf{r}_j^\top \omega} \sqrt{\frac{g(\omega)}{p(\omega)}}\ .$$

In the mean time, by definition of φ and ψ, we have

$$\varphi(\mathsf{r}_i)^\top \psi(\mathsf{r}_j) = \frac{1}{r} \sum_{k=1} e^{2\pi \mathsf{i} \mathsf{r}_i^\top \omega_k} \sqrt{\frac{g(\omega_k)}{p(\omega_k)}} \cdot e^{-2\pi \mathsf{i} \mathsf{r}_j^\top \omega_k} \sqrt{\frac{g(\omega_k)}{p(\omega_k)}}$$

Finally, note that $\omega_1, \cdots, \omega_m \sim$ i.i.d. p. By linearity of expectation, we have $f(\mathsf{r}_i - \mathsf{r}_j) = \mathsf{E}\ \varphi(\mathsf{r}_i)^\top \psi(\mathsf{r}_j)$ and conclude the proof. □

A.2 Variance of RPE approximation

Lemma A.2. *Assume that $c = \||g(\mathsf{x})|/p(\mathsf{x})\|_\infty$, where g is the Fourier Transform of the RPE function f and p is the probability density function of some probabilistic distribution. Then the following is true for any $\mathsf{z} \in \mathsf{R}^\ell$:*

$$\mathrm{Var}_{\omega\sim p}\ e^{2\pi \mathsf{i} \mathsf{z}^\top \omega} \frac{g(\omega)}{p(\omega)} \le c^2 - f(\mathsf{z})^2. \tag{6}$$

[4] This is Theorem 4.1 in the main paper.

Proof. Recall that for a complex random variable Z, its variance is defined as

$$\mathrm{Var}[Z] = \mathbb{E}\left[(Z - \mathbb{E}[Z])(Z - \mathbb{E}[Z])^*\right] = \mathbb{E}[ZZ^*] - \mathbb{E}[Z]\,\mathbb{E}[Z]^*, \tag{7}$$

where * denotes the conjugate.

Straightforward calculation gives

$$\mathbb{E}\left[e^{2\pi i \mathbf{z}^\top } \frac{g( )}{p( )} \right] \left[e^{2\pi i \mathbf{z}^\top } \frac{g( )}{p( )} \right]^* = \mathbb{E}\left[e^{2\pi i \mathbf{z}^\top } \frac{g( )}{p( )} \cdot e^{-2\pi i \mathbf{z}^\top } \frac{g( )^*}{p( )} \right] \tag{8}$$

$$= \mathbb{E}\left[\frac{|g( )|^2}{p( )^2} \right] \le c^2. \tag{9}$$

Besides, Eq. (5) implies that

$$\mathbb{E}\left[e^{2\pi i \mathbf{z}^\top } \frac{g( )}{p( )} \right] = f(\mathbf{z}). \tag{10}$$

Plugging the above two results into Eq. (7) yields Eq. (6) and hence concludes the proof. □

Theorem A.3 (Variance of RPE approximation)**.** *Under the assumption of Lemma A.2, for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^\ell$, the variance of the approximation given by $\varphi(\mathbf{x})^\top \psi(\mathbf{y})$ in Theorem A.1 satisfies*

$$\mathrm{Var}[\varphi(\mathbf{x})^\top \psi(\mathbf{y})] \le \frac{c^2 - f(\mathbf{x}-\mathbf{y})^2}{r}. \tag{11}$$

Proof. Note that

$$\varphi(\mathbf{x})^\top \psi(\mathbf{y}) = \frac{1}{r} \sum_{t=1} e^{2\pi i (\mathbf{x}-\mathbf{y})^\top \ t} \frac{g(\ t)}{p(\ t)}, \tag{12}$$

where the random features $ _1, \cdots, _r$ are r i.i.d. samples from the distribution p.

Setting $\mathbf{z} = \mathbf{x} - \mathbf{y}$ in Eq. (6) and considering r i.i.d. samples immediately yield Eq. (11). □

A.3 Uniform convergence and sample complexity of RPE approximation

Theorem A.4 (Uniform convergence and sample complexity for approximation)**.** [5] *Given L vectors $\mathbf{r}_1, ..., \mathbf{r}_L \in \mathbb{R}^\ell$, define the RPE attention mask $\mathbf{N} = [f(\mathbf{r}_i - \mathbf{r}_j)]_{i,j\in[L]}$. Assume that $c = \| |g(\mathbf{x})|/p(\mathbf{x}) \|_\infty$, where g is the Fourier Transform of f and p is some probability density function over $\mathbb{R}^\ell$.*

For any $\varepsilon, \delta > 0$, if the number of random features $r = \Theta\left(\frac{c^2}{\varepsilon^2} \log \frac{L}{\delta} \right)$, then FLT*'s RPE approximator $\hat{\mathbf{N}}$ satisfies*

$$\mathbb{P}\left[\|\mathbf{N} - \hat{\mathbf{N}}\|_{\max} \le \varepsilon \right] > 1 - \delta,$$

where $\|\cdot\|_{\max}$ denotes the max norm of a matrix.

Proof. For any $i, j \in \{1, \cdots, L\}$, the assumption $c = \| |g(\mathbf{x})|/p(\mathbf{x}) \|_\infty$ implies that almost surely

$$e^{2\pi i (\mathbf{r}_i - \mathbf{r}_j)^\top \ t} \frac{g(\ t)}{p(\ t)} \le c \qquad (\forall\, t \in \{1, \cdots, r\}), \tag{13}$$

where $ _1, \cdots, _r$ denote the r random features from the distribution p.

Note that Eq. (13) implies that both the real part and imaginary part of the estimated RPE is bounded. Applying Hoeffding Inequality (to the real part and imaginary part) yields

$$\mathbb{P}\left[\frac{1}{t} \sum_{t=1} e^{2\pi i (\mathbf{r}_i - \mathbf{r}_j)^\top \ t} \frac{g(\ t)}{p(\ t)} - f(\mathbf{r}_i - \mathbf{r}_j) > \varepsilon \right] < 4 e^{-\frac{r \varepsilon^2}{4c^2}}. \tag{14}$$

[5] This is Theorem 4.2 in the main paper.

By union bound over $i, j \in \{1, \cdots, L\}$, we have

$$\mathsf{P}\left[\exists\, i,j \in \{1,\cdots,L\},\ \text{s.t.}\ \ \frac{1}{t}\sum_{t=1}^{\mathsf{X}} e^{2\pi \mathrm{i}(\mathbf{r}_i-\mathbf{r}_j)^\top \ t}\frac{g(\ t)}{p(\ t)} - f(\mathbf{r}_i-\mathbf{r}_j)\ > \varepsilon\right] < 4L^2 e^{-\frac{r\varepsilon^2}{4c^2}}. \tag{15}$$

Equivalently, with probability at least $4L^2 e^{-\frac{r\varepsilon^2}{4c^2}}$, we have

$$\frac{1}{r}\sum_{t=1}^{\mathsf{X}} e^{2\pi \mathrm{i}(\mathbf{r}_i-\mathbf{r}_j)^\top \ t}\frac{g(\ t)}{p(\ t)} - f(\mathbf{r}_i-\mathbf{r}_j)\ > \varepsilon \qquad (\forall\, i,j \in \{1,\cdots,L\}) \tag{16}$$

$$\Rightarrow \qquad \|\mathsf{N} - \hat{\mathsf{N}}\|_{\max} \leq \varepsilon. \tag{17}$$

Set

$$r = \frac{4c^2}{\varepsilon^2}\log\frac{4L^2}{\delta} = \Theta\left(\frac{c^2}{\varepsilon^2}\log\frac{L}{\delta}\right). \tag{18}$$

Then we have

$$\mathsf{P}\left[\|\mathsf{N} - \hat{\mathsf{N}}\|_{\max} \leq \varepsilon\right] > 1 - 4L^2 e^{-\frac{r\varepsilon^2}{4c^2}} = 1-\delta, \tag{19}$$

which concludes the proof. □

A.4 Discussions on the theoretical results

Theorem A.4 implies that FLT's estimated RPE mask $\hat{\mathsf{N}}$ can approximate the true RPE mask N up to arbitrary precision with high probability. Besides, note that the constant c can be viewed as fixed for a pretrained model which does not depend on L. Thus, in order to obtain an arbitrarily accurate RPE approximator, the required number of random features only scales *logarithmically* with L. This property is particularly appealing because it indicates FLTcan remain accurate in the long sequence regimes while accelerating powerful RPE-enhanced attention.

Theorems A.3 & A.4 also provide insights on finding the optimal p. Note that the variance and the sample complexity both scale with $c = \||g(\mathbf{x})|/p(\mathbf{x})\|_\infty$, and lower c can potentially leads to better approximation. Specifically, choosing $p(\mathbf{x}) \propto |g(\mathbf{x})|$ minimizes c under the constraint of p being a probability density function. The shift-invariant kernels RPE indeed satisfies this property, and is optimal in terms of the approximation sample complexity.

Another simple yet effective approach is to parameterize p as a Gaussian distribution with *learnable* means and variances. The optimization procedure can search for the optimal p in the class of Gaussian distributions to obtain good RPE approximation. This technique turns out to be helpful for the experiment on molecular property prediction (Sec. 5.3 in the main paper).

Finally, we point out that the $\log L$ factor in the sample complexity bound in Theorems A.4 is introduced for technical reasons: the convergence analysis is conducted for the random features applying exponential mapping which is not bounded.[6] That being said, this logarithmic factor is still negligible (as opposed to polynoimal dependency).

B DETAILED EXPERIMENT SETTINGS

All tested Transformer variants were trained and tested on a TPU pods containing 4 TPU v3 chips with JAX and on GPUs (V100).

B.1 Language modeling

In this experiment, we study FLT with two RPE variants, Gaussian mixture RPE and local RPE. The detailed descriptions of baselines have been provided in the main paper.

[6] For random features applying trigonometric functions and thus could leverage "-net trick combined with strong Lipschitz function argument. An analogous result can be found in [Choromanski et al., 2021]

All the tested models are efficient Transformers based on kernelized low-rank attention, with 6 decoder layers. In each layer, there are 8 attention heads. The hidden dimension is set to 512. The dimension of the feed-forward sub-layer is set to 2048. The feature map dimension m is set to 64 in the low-rank approximation of the attention matrix. For our FLT models, the number of random features for RPE r is set to 32. We use the validation perplexity as the evaluation metric: lower perplexity indicates better performances.

Following existing works [Peng et al., 2021, Luo et al., 2021], the sequence length is set to 512 during both training and evaluation. All models are trained *without* access to the context from previous mini-batches for a fair comparison. The dropout ratio and weight decay are set to 0.1 and 0.01, respectively. The batch size is set to 64. We use Adam as the optimizer, and set its hyperparameter ε to $1\mathrm{e}-6$ and (β_1, β_2) to (0.9, 0.98). The model is trained for 150k steps with a 6k-step warm-up stage followed by an inverse square-root learning rate scheduler, with the peak learning rate set to $2\mathrm{e}-3$.

For the FLT variant with Gaussian mixture RPE, the FT of the RPE function, i.e., the function g, is parameterized as Eq. (4.3):

$$g(\ ) = \sum_{t=1} w_t \exp\ -\frac{\| \ - \ _t\|^2}{2\sigma_t^2}\ .$$

For the FLT variant with local RPE, the function g is parameterized as

$$g(\xi) = \sum_{t=1} w_t \cdot \frac{\sin(2\pi v_t \xi)}{\pi \xi}, \tag{20}$$

where $w_1, \cdots, w_T$ and $v_1, \cdots, v_T$ are learnable parameters and T is a pre-defined hyper-parameter. In this case, the underlying implict RPE function f is

$$f(\Delta r) = \sum_{t=1} w_t \cdot \mathsf{I}[|\Delta r| \leq v_t]. \tag{21}$$

For both FLT variants, the RPE masks are different in different attention heads, but are *shared* across different layers. The random features $\xi_1, \cdots, \xi_r$ are sampled from the standard Gaussian distribution.

B.2 Image classification

Table 4 presents the basic statistics of the datasets used in the image classification experiements.

Table 4: Details of the datasets used in image classification tasks with the FourierLearner-Transformer.

Dataset name	# of classes	Training set size	Test set size
ImageNet2012 [Deng et al., 2009]	1K	1.2M	100K
Places365 [Zhou et al., 2018]	365	1.8M	328K
Fashion-MNIST [Xiao et al., 2017]	10	60K	10K

All tested models consist of 12 layers with 12 attention heads in each layer. The dimension of the feed-forward sub-layer is set to 3072. In our FLT, we use learnable ReLU as the feature map for kernelized linear attention. In particular, the feature map is $\phi : \mathsf{x} \mapsto \mathsf{W}\mathsf{x}$ where W is a learnable matrix. For all the models, we used a dropout rate of 0.1 and no attention dropout. We applied the Adam optimizer with weight decay equal to 0.05 and a standard batch size of 4096. All Transformers were trained on TPU architectures until convergence.

B.3 Molecular property prediction

We adopt most of the training strategies of 3D-Graphormer [Shi et al., 2022]. Specifically, we trained a regular Performer with 12 layers and two FLT with 10 and 12 layers respectively. Following existing works [Jumper et al., 2021, Shi et al., 2022], model outputs are repeatedly fed to the model for four times. In each layer,

Table 5: Hyperparameters for Image Classification.

Parameter	Value
Batch size	4096
Optimizer	AdamW
Base Learning rate	$1.5\text{e}-4$
Weight decay	0.05
Optimizer momentum	$(\beta_1, \beta_2) = (0.9, 0.95)$
Learning rate schedule	cosine decay
Warm up epochs	40
Augmentation	RandomResizedCrop
Compute resources	8×8 TPUv3

there are 48 attention heads. The hidden dimension is set to 768. The dimension of the feed-forward sub-layer is set to 2048. The feature map dimension m is set to 64 in the low-rank approximation of the attention matrix. For our FLT models, the number of random features for RPE r is set to 16 and the number of Gaussian basis functions in RPE T is set to 32. The random feature ξ_i are sampled from Gaussian distribution $\mathcal{N}(0, \sigma_i^2 \mathbf{I})$, where σ_i is learnable.

We evaluate the performance of the tested models on the in-domain validation set, where the validation samples come from the same distribution as the training distribution. We use Mean Absolute Error (MAE) of the energies and the percentage of Energies within a Threshold (EwT) of the ground truth energy to evaluate the accuracy of the predicted energies.

For all the models, the attention dropout ratio and the weight decay are set to 0.1 and 0.001, respectively. The batch size is set to 64. We use Adam as the optimizer, and set its hyperparameter ε to $1\text{e}-6$ and (β_1, β_2) to (0.9, 0.98). The peak learning rate is set to $3\text{e}-4$ with a 10K-step warm-up stage. After the warm-up stage, the learning rate decays linearly to zero. All the models are trained for 500k steps in total.

B.4 Learnable optimizers

In all the experiments, meta-training pipelines and the training recipes from [Jain et al., 2023] are applied. Following [Jain et al., 2023], all learnable models are used as memory units in the corresponding learnable optimizers and meta-trained in the exact same way on a small set of unrelated optimization tasks. Furthermore, all attention-based memory mechanisms are derived from [Jain et al., 2023].

C VISUALIZATIONS

C.1 Local RPEs

In Fig. 4, we visualize the shape of local RPEs that can be modeled with FLTs via Fourier Transform.

C.2 Attention matrices

In Fig. 5, we visualize the attention matrices of an FLT model trained on WikiText-103 language modeling. In particular, we feed one sequence in the training set as the input to the model and visualize the attention matrices of the 8 attention heads in the first layer. It can be seen that some attention heads pay more attention to nearby tokens, while others shows global attention patterns. The average attention probability over the most distant/nearby 10% tokens is 0.068/0.279 respectively. This result shows that FLT learns locality bias in language while maintaining the advantage to capture global contexts and leverage information in distant tokens.