UNCOVERING HIDDEN GEOMETRY IN TRANSFORMERS VIA DISENTANGLING POSITION AND CONTEXT

Anonymous authors
Paper under double-blind review

ABSTRACT

Transformers are widely used to extract complex semantic meanings from input tokens, yet they usually operate as black-box models. In this paper, we present a simple yet informative decomposition of hidden states (or embeddings) of trained transformers into interpretable components. For any layer, embedding vectors of input sequence samples are represented by a tensor $\boldsymbol{h} \in \mathbb{R}^{C \times T \times d}$. Given embedding vector $\boldsymbol{h}_{c,t} \in \mathbb{R}^d$ at sequence position $t \leq T$ in a sequence (or context) $c \leq C$, extracting the mean effects yields the decomposition

$$\boldsymbol{h}_{c,t} = \boldsymbol{\mu} + \mathbf{pos}_t + \mathbf{ctx}_c + \mathbf{resid}_{c,t}$$

where $\boldsymbol{\mu}$ is the global mean vector, $\mathbf{pos}_t$ and $\mathbf{ctx}_c$ are the mean vectors across contexts and across positions respectively, and $\mathbf{resid}_{c,t}$ is the residual vector. For popular transformer architectures and diverse text datasets, empirically we find pervasive mathematical structure: (1) $(\mathbf{pos}_t)_t$ forms a low-dimensional, continuous, and often spiral shape across layers, (2) $(\mathbf{ctx}_c)_c$ shows clear cluster structure that falls into context topics, and (3) $(\mathbf{pos}_t)_t$ and $(\mathbf{ctx}_c)_c$ are mutually incoherent—namely $\mathbf{pos}_t$ is almost orthogonal to $\mathbf{ctx}_c$—which is canonical in compressed sensing and dictionary learning. This decomposition offers structural insights about input formats in in-context learning (especially for induction heads) and in arithmetic tasks.

1 INTRODUCTION

Transformers (Vaswani et al., 2017) are practical neural network models that underlie recent successes of large language models (Brown et al., 2020; Bubeck et al., 2023). Unfortunately, transformers are often used as black-box models due to lack of in-depth analyses of internal mechanism, which raises concerns such as lack of interpretability, model biases, security issues, etc., (Bommasani et al., 2021).

In particular, it is poorly understood what information embeddings from each layer capture. We identify two desiderata: (1) internal quantitative measurements, particularly for the intermediate layers; (2) visualization tools and diagnostics tailored to transformers beyond attention matrix plots.

Let us introduce basic notations. An input sequence consists of T consecutive tokens (e.g., words or subwords), and a corpus is a collection of all input sequences. Let C be the total number of input sequences and $c \leq C$ denote a generic sequence, which may be represented by $\boldsymbol{x}_{c,1}, \ldots, \boldsymbol{x}_{c,T}$ where each $\boldsymbol{x}_{c,t}$ corresponds to a token. We start from the initial static (and positional) embeddings $(\boldsymbol{h}_{c,t}^{(0)})_{t \leq T}$ and then calculate the intermediate-layer embeddings $(\boldsymbol{h}_{c,t}^{(\ell)})_{t \leq T}$:

$$\begin{aligned} &\boldsymbol{h}_{c,1}^{(0)}, \ldots, \boldsymbol{h}_{c,T}^{(0)} = \texttt{Embed}(\boldsymbol{x}_{c,1}, \ldots, \boldsymbol{x}_{c,T}) \\ &\boldsymbol{h}_{c,1}^{(\ell)}, \ldots, \boldsymbol{h}_{c,T}^{(\ell)} = \texttt{TFLayer}_\ell(\boldsymbol{h}_{c,1}^{(\ell-1)}, \ldots, \boldsymbol{h}_{c,T}^{(\ell-1)}) \qquad \text{for } \ell = 1, \ldots, L. \end{aligned}$$

where $\texttt{Embed}$ and $\texttt{TFLayer}_\ell$ are general mappings. This general definition encompasses many transformer models, which depend on attention heads defined as follows. Given $d_{\text{head}} \leq d$ and input matrix $\boldsymbol{X} \in \mathbb{R}^{T \times d}$, for trainable weights $\boldsymbol{W}^q, \boldsymbol{W}^k, \boldsymbol{W}^v \in \mathbb{R}^{d \times d_{\text{head}}}$, define

$$\mathrm{AttnHead}(\boldsymbol{X}) = \mathrm{softmax}\left(\frac{\boldsymbol{X}\boldsymbol{W}^q(\boldsymbol{W}^k)^\top \boldsymbol{X}^\top}{\sqrt{d_{\text{head}}}}\right)\boldsymbol{X}\boldsymbol{W}^v \in \mathbb{R}^{T \times d_{\text{head}}}. \tag{1}$$

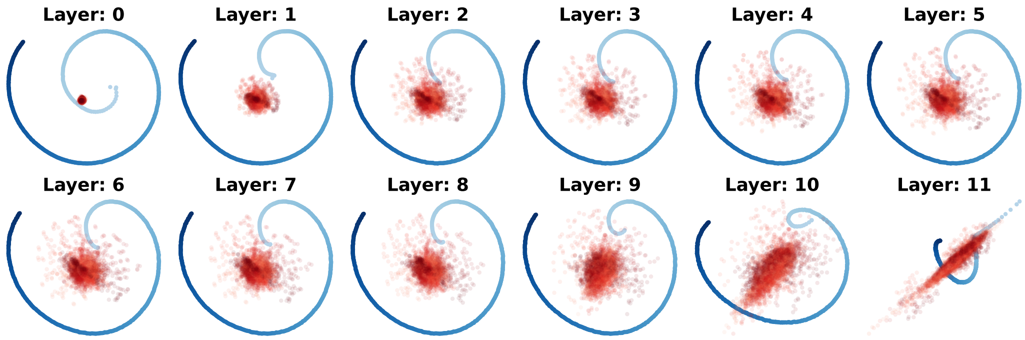

Figure 1: PCA visualization of positional basis (blue) and cvecs (red) from GPT-2 on OpenWebText. For every layer ℓ, each $\mathbf{pos}_t^{(\ell)}$ and randomly selected $\mathbf{cvec}_{c,t}^{(\ell)}$ are projected using top-2 principal directions of $(\mathbf{pos}_t^{(\ell)})_{t\le T}$. Darker blue/red colors correspond to larger t. Principal components have different scales across layers, but for aesthetic purposes we rescaled all plots.

Multi-head attention heads, denoted by MHA, are essentially the concatenation of many attention heads. Denote a generic fully-connected layer by $\texttt{FFN}(\boldsymbol{x}) = \boldsymbol{W}_2 \max\{\mathbf{0}, \boldsymbol{W}_1\boldsymbol{x} + \boldsymbol{b}_1\} + \boldsymbol{b}_2$ given any $\boldsymbol{x} \in \mathbb{R}^d$ for trainable weights $\boldsymbol{W}_1 \in \mathbb{R}^{d' \times d}, \boldsymbol{W}_2 \in \mathbb{R}^{d\times d'}, \boldsymbol{b}_1, \boldsymbol{b}_2 \in \mathbb{R}^d$ (often $d' = 4d$), and let LN be a generic layer normalization layer. The standard transformer is expressed as

$$\boldsymbol{h}_c^{(\ell+0.5)} = \boldsymbol{h}_c^{(\ell)} + \texttt{MHA}^{(\ell)}(\texttt{LN}^{(\ell,1)}(\boldsymbol{h}_c^{(\ell)})), \quad \boldsymbol{h}_{c,t}^{(\ell+1)} = \boldsymbol{h}_{c,t}^{(\ell+0.5)} + \texttt{FFN}^{(\ell)}(\texttt{LN}^{(\ell,2)}((\boldsymbol{h}_{c,t}^{(\ell+0.5)})))$$

where $\boldsymbol{h}_c^{(\ell+0.5)} = (\boldsymbol{h}_{c,1}^{(\ell+0.5)}, \ldots, \boldsymbol{h}_{c,T}^{(\ell+0.5)})$ and $\boldsymbol{h}_c^{(\ell)} = (\boldsymbol{h}_{c,1}^{(\ell)}, \ldots, \boldsymbol{h}_{c,T}^{(\ell)})$.

1.1 A MEAN-BASED DECOMPOSITION

For each embedding vector $\boldsymbol{h}_{c,t}^{(\ell)} \in \mathbb{R}^d$ from any trained transformer, consider the decomposition

$$\boldsymbol{h}_{c,t}^{(\ell)} = \boldsymbol{\mu}^{(\ell)} + \mathbf{pos}_t^{(\ell)} + \mathbf{ctx}_c^{(\ell)} + \mathbf{resid}_{c,t}^{(\ell)}, \qquad \text{where} \tag{2}$$

$$\boldsymbol{\mu}^{(\ell)} := \frac{1}{CT}\sum_{c,t} \boldsymbol{h}_{c,t}^{(\ell)}, \quad \mathbf{pos}_t^{(\ell)} := \frac{1}{C}\sum_c \boldsymbol{h}_{c,t}^{(\ell)} - \boldsymbol{\mu}^{(\ell)}, \quad \mathbf{ctx}_c^{(\ell)} := \frac{1}{T}\sum_t \boldsymbol{h}_{c,t}^{(\ell)} - \boldsymbol{\mu}^{(\ell)}. \tag{3}$$

Each of the four components has the following interpretations. For any given layer ℓ,

- we call $\boldsymbol{\mu}^{(\ell)}$ the global mean vector, which differentiates neither contexts nor positions;
- we call $(\mathbf{pos}_t^{(\ell)})_{t\le T}$ the positional basis, as they quantify average positional effects;
- we call $(\mathbf{ctx}_c^{(\ell)})_{c\le C}$ the context basis, as they quantify average sequence/context effects;
- we call $(\mathbf{resid}_{c,t}^{(\ell)})_{t\le T, c\le C}$ the residual vectors, which capture higher-order effects.
- In addition, we define $\mathbf{cvec}_{c,t}^{(\ell)} = \mathbf{ctx}_c^{(\ell)} + \mathbf{resid}_{c,t}^{(\ell)}$.

A priori, we do not know how much position information is retained in each layer, since many transformers only have explicit positional encodings in the 0-th layer. Are positional basis and context basis play the role as the names suggest? We will provide affirmative answers.

Sampling input sequences. A corpus can be extremely large, containing billions of tokens. For practical use, in this paper C is much smaller: we subsample input sequences from the corpus; for example, $C = 6.4K$ in Figure 1. Thus, our empirical-mean-based decomposition can be regarded as an estimate of the population means, using much less computation.

On terminology. (i) We use *context* to refer to a sequence since its tokens collectively encode context information. (ii) We call positional/context basis for convenience. A more accurate term is *frame* or *overcomplete basis*, since $(\mathbf{pos}_t^{(\ell)})_{t\le T}$ and $(\mathbf{ctx}_t^{(\ell)})_{c\le C}$ are often linearly dependent.

Table 1: **Averaged (and std of) measurements across layers.** Measurements based on 6.4K samples. All values are in $[0, 1]$ except 'rank estimate': 'relative norm' means magnitude of positional basis relative to centered embeddings; 'similarity' and 'incoh' are averaged *cosine similarity* (inner products of normalized vectors) between $\mathbf{ctx}$, and between $\mathbf{ctx}$ and $\mathbf{pos}$, respectively.

		Positional basis		Context basis		Incoh
		rank estimate	relative norm	inter-cluster similarity	intra-cluster similarity	
NanoGPT	Shakespeare	7.86(1.96)	0.66(0.28)	——	——	——
GPT-2	OpenWebText	11.38(1.86)	0.31(0.19)	0.10(0.01)	0.44(0.04)	0.051(0.05)
	WikiText	11.69(1.64)	0.31(0.19)	0.11(0.01)	0.41(0.03)	0.039(0.04)
BERT	OpenWebText	12.54(2.73)	0.24(0.07)	0.13(0.04)	0.26(0.04)	0.046(0.05)
	WikiText	12.62(2.70)	0.24(0.06)	0.17(0.03)	0.31(0.04)	0.043(0.04)
BLOOM	OpenWebText	10.23(1.31)	0.16(0.09)	0.21(0.12)	0.48(0.06)	0.158(0.23)
	WikiText	10.00(1.47)	0.16(0.08)	0.15(0.14)	0.32(0.09)	0.148(0.23)
Llama 2	OpenWebText	9.38(1.15)	0.14(0.03)	0.12(0.13)	1.00(0.01)	0.190(0.24)
	WikiText	8.69(0.91)	0.14(0.03)	0.33(0.20)	1.00(0.01)	0.316(0.27)
	GitHub	8.69(1.67)	0.20(0.05)	0.22(0.08)	1.00(0.01)	0.189(0.20)

Connections to Analysis-of-Variance (ANOVA). Our embedding decomposition is similar to two-way ANOVA in form. Borrowing standard terminology from ANOVA, positions and contexts can be regarded as two *treatments*, so viewing the embedding $\boldsymbol{h}_{c,t}$ as the response variable, then positional/context bases represent mean effects.

1.2 Pervasive geometrical structure

We consider a variety of transformers and datasets; see Section A for details. Our main results are summarized below. Further, Section E explores randomization experiments and arithmetic tasks.

1. Positional basis is a significant and approximately low-rank component, forming a continuous and curving shape, which is linked to smoothness.
2. Context basis has strong cluster patterns corresponding to documents/topics.
3. Positional basis and context basis are nearly orthogonal (or *incoherent*), which allows self-attention heads to capture the interaction of the two bases easily.

What does $\mathbf{resid}_{c,t}$ represent? As with regression models, residual components may be non-negligible and contain idiosyncratic information. For example, they can be used to track previously seen tokens (Section 4.1) or special symbols in arithmetic (Section 5.2).

2 Geometry of positional basis

2.1 Low-dimensional structure as a significant component

We find that the positional basis concentrates around a low-dimensional subspace. In Table 1, we report the rank estimate of positional basis averaged across all layers using the method of Donoho et al. (2023). In Section B.2, we report detailed rank estimates and an additional measurement: stable rank (Rudelson & Vershynin, 2007). Table 2 shows that the low-rank structure is robustness to out-of-distribution data, suggesting positional basis is indeed agnostic to contexts.

We also find that usually, the positional basis accounts for a significant proportion of embeddings. In Table 1, we report the relative norm (averaged across layers) $\|\boldsymbol{P}\|_{\mathrm{op}}/\|\boldsymbol{M}\|_{\mathrm{op}}$, where $\boldsymbol{M}$ contains centered embedding vectors $\boldsymbol{h}_{c,t} - \boldsymbol{\mu}$ and columns of $\boldsymbol{P}$ are corresponding $\mathbf{pos}_t$. We also consider normalized vectors: $\bar{\boldsymbol{P}} = [\frac{\mathbf{pos}_1}{\|\mathbf{pos}_1\|}, \dots, \frac{\mathbf{pos}_t}{\|\mathbf{pos}_t\|}]$, etc. In Figure 2 (left), we plot the top singular values (adjusted for dimensional difference) in descending order of $\boldsymbol{P} = [\mathbf{pos}_1, \dots, \mathbf{pos}_T]$, $\boldsymbol{Cvec} = [\mathbf{cvec}_{1,1}, \dots, \mathbf{cvec}_{c,T}]$, $\boldsymbol{R} = [\mathbf{resid}_{1,1}, \dots, \mathbf{resid}_{c,T}]$. Visibly, positional basis is a considerable component in magnitude and contributes to the spikedness of embeddings.

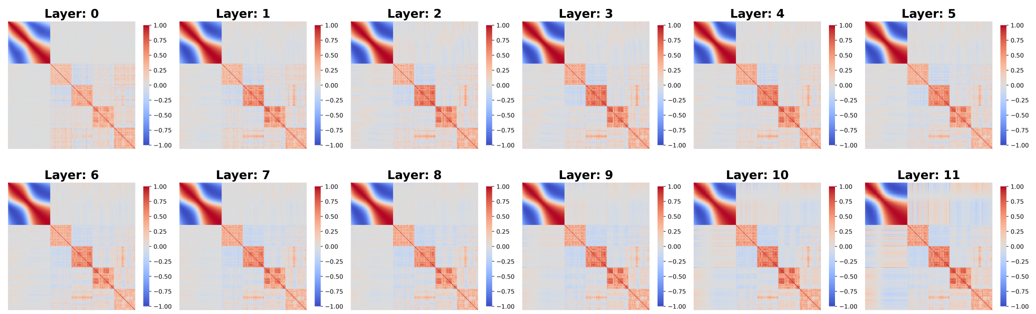

Figure 3: **Normalized Gram matrix** $[\bar{\boldsymbol{P}}, \bar{\boldsymbol{C}}]^\top [\bar{\boldsymbol{P}}, \bar{\boldsymbol{C}}]$ where $\bar{\boldsymbol{P}} = [\frac{\mathbf{pos}_1}{\|\mathbf{pos}_1\|}, \ldots, \frac{\mathbf{pos}_t}{\mathbf{pos}_T}]$ and $\bar{\boldsymbol{C}} = [\frac{\mathbf{ctx}_1}{\|\mathbf{ctx}_1\|}, \ldots, \frac{\mathbf{ctx}_C}{\|\mathbf{ctx}_C\|}]$ based on GPT-2. Here, $T = 128$, and $\mathbf{ctx}_c$ is sampled from 4 documents with sample size 32 in OpenWebText. We find (i) **Smoothness**, $\mathbf{pos}$-$\mathbf{pos}$ part (top left) of Gram matrix is smooth; (ii) **Incoherence**, $\mathbf{pos}$-$\mathbf{ctx}$ part (top right/bottom left) has values close to 0; (iii) **Clustering**, $\mathbf{ctx}$-$\mathbf{ctx}$ part (bottom right) shows strong cluster patterns.

We notice that there are two exceptions: (i) 0-th layer of Llama 2 and BLOOM (due to no positional encoding), (ii) last one/few layers of a transformer. Likely, last layers do not need position information as contextualization is completed; an investigation is left as future work.

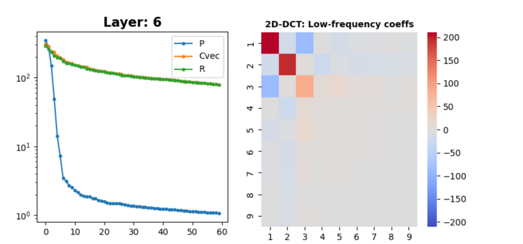

Figure 2: **Spectral and Fourier analysis** based on GPT-2 model and OpenWebText. **Left:** Top-60 (adjusted) singular values of $\boldsymbol{P}, \boldsymbol{Cvec}, \boldsymbol{R}$. **Right:** Applying 2D discrete cosine transform to $\bar{\boldsymbol{P}}^\top \bar{\boldsymbol{P}}$, we show first 10 frequency coefficients.

2.2 Spiral shape via a Fourier perspective

A priori, a common geometric structure of positional basis is unexpected: after all, different models/datasets may use position information differently. Nevertheless, on text-based datasets, we observe a common continuous shape that is often spiral, parabolic, or U-shaped.

In Figure 2 (right), we apply the 2D discrete cosine transform to the normalized Gram matrix $\bar{\boldsymbol{P}}^\top \bar{\boldsymbol{P}}$ and discover that energies are concentrated mostly in the low-frequency components, which reinforces the smooth and curving structure we identified. Other models show similar low-frequency patterns except for BERT.

Table 2: **Robustness of positional basis**. Similar geometric structures found on *out-of-distribution* samples: NanoGPT on WikiText, others on GitHub.

	rank estimate	relative norm
NanoGPT	7.86(1.96)	0.66(0.28)
GPT-2	11.54(1.55)	0.31(0.16)
BERT	12.46(2.47)	0.23(0.04)
BLOOM	9.54(0.84)	0.14(0.05)

2.3 Theoretical insight: connection to smoothness

It is well known that the smoothness of a function is connected to fast decay or sparsity in the frequency domain (Pinsky, 2008, Sect. 1.2.3). In Figure 3, the Gram matrix of positional basis exhibits smooth patterns, allowing attention to neighboring tokens more easily. Does smoothness of the Gram matrix shed light on the geometrical structure of the positional basis? We provide an affirmative answer.

Smoothness of $\mathbf{pos}$-$\mathbf{pos}$ *Gram matrix induces the low-dimensional and spiral shape.*

Let $\boldsymbol{G} = \boldsymbol{P}^\top \boldsymbol{P} \in \mathbb{R}^{T \times T}$ be the Gram matrix of the positional basis (no normalization for simplicity). By definition in Equation 3, positional basis has zero means, so $\mathbf{pos}_1 + \ldots + \mathbf{pos}_T = \mathbf{0}$. To characterize smoothness, below we introduce the definition of finite difference. As with the discrete cosine transform in 1D, we need to extend and reflect the Gram matrix to avoid boundary effects.

Let $\boldsymbol{G}^{(1)} = \boldsymbol{G}$ and $\boldsymbol{G}^{(2)}, \boldsymbol{G}^{(3)}, \boldsymbol{G}^{(4)} \in \mathbb{R}^{T\times T}$ be defined by $\boldsymbol{G}^{(2)}_{t,t'} = \boldsymbol{G}_{t,T+1-t'}$, $\boldsymbol{G}^{(3)}_{t,t'} = \boldsymbol{G}_{T+1-t,t'}$, $\boldsymbol{G}^{(4)}_{t,t'} = \boldsymbol{G}_{T+1-t,T+1-t'}$ for any $t, t' = 1, 2, \dots T$. We extend and reflect $\boldsymbol{G}$ by

$$\tilde{\boldsymbol{G}} := \begin{pmatrix} \boldsymbol{G}^{(1)} & \boldsymbol{G}^{(2)} \\ \boldsymbol{G}^{(3)} & \boldsymbol{G}^{(4)} \end{pmatrix}. \tag{4}$$

We define the first-order finite difference by (using periodic extension $\tilde{G}_{t\pm 2T,t'\pm 2T} = \tilde{G}_{t,t'}$)

$$[\Delta^{(1,1)}\tilde{\boldsymbol{G}}]_{t,t'} = T^2\big(\tilde{G}_{t,t'} - \tilde{G}_{t-1,t'} - \tilde{G}_{t,t'-1} + \tilde{G}_{t-1,t'-1}\big), \qquad \text{for all integers } t, t' \tag{5}$$

Higher-order finite differences are defined recursively by $\Delta^{(m,m)}\tilde{\boldsymbol{G}} = \Delta^{(1,1)}\big(\Delta^{(m-1,m-1)}\tilde{\boldsymbol{G}}\big)$.

Note that $\Delta^{(m,m)}\tilde{\boldsymbol{G}}$ measures higher-order smoothness of $\tilde{\boldsymbol{G}}$. Indeed, if $G_{t,t'} = f(t/T, t'/T)$ for certain smooth function $f(x,y)$ defined on $[0,1]^2$, then $[\Delta^{(m,m)}\tilde{\boldsymbol{G}}]_{t,t'} \approx \partial_x^m \partial_y^m f(t/T, t'/T)$.

Theorem 1. *Fix positive integers $k \le T$ and m. Define the low-frequency vector $\boldsymbol{f}_s = (1, \cos((s-0.5)\pi/T), \dots, \cos((s-0.5)(T-1)\pi/T))^\top \in \mathbb{R}^T$ where $s = 1, \dots, k$, and denote $\boldsymbol{F}_{\le k} = [\boldsymbol{f}_1, \dots, \boldsymbol{f}_k] \in \mathbb{R}^{T\times k}$. Then there exists $\boldsymbol{B} \in \mathbb{R}^{k\times k}$ such that (denoting $\|\boldsymbol{A}\|_{\max} = \max_{ij}|A_{ij}|$)*

$$\frac{1}{T}\left\|\boldsymbol{G} - (\boldsymbol{F}_{\le k}\boldsymbol{B})^\top \boldsymbol{F}_{\le k}\boldsymbol{B}\right\|_{\mathrm{op}} \le \frac{6}{(8k)^m}\|\Delta^{(m,m)}\tilde{\boldsymbol{G}}\|_{\max}. \tag{6}$$

This theorem implies that if the extended Gram matrix has higher-order smoothness, namely $\|\Delta^{(m,m)}\tilde{\boldsymbol{G}}\|_{\max}$ is bounded by a constant, then even for moderate k and m, we have approximation $\boldsymbol{G} \approx (\boldsymbol{F}_{\le k}\boldsymbol{B})^\top \boldsymbol{F}_{\le k}\boldsymbol{B}$. Note that $\boldsymbol{F}_{\le k}\boldsymbol{B}$ consists of linear combinations of low-frequency vectors. This explains why $\boldsymbol{G}$ has a dominant low-rank and low-frequency component.

Why smoothness? One possible explanation is that smoothness allows attention to neighboring tokens easily (crucial for natural languages/codes), because often short-ranged token pairs tend to receive higher QK values and thus higher attention weights.

3 Context basis: salient cluster structure

Figure 3 shows that the context basis computed from 4 different documents can be visually clustered into 4 groups. Measuring cluster compactness, we find that using context basis or even cvecs has at least a slight advantage over raw embeddings (without removing positional effects) as in (Thompson & Mimno, 2020). See Section C for detailed analysis.

On contextualization. We observe from Table 1: (i) except for Llama 2 and BLOOM, the increase in cluster compactness seems to be moderate and only occurs in early layers, likely because our measurements are global rather than based on fine-grained conditional probabilities; (ii) Llama 2 and BLOOM show progressive changes in cluster compactness across layers, as shown by the numbers in the paratheses. This is likely due to heterogeneous data during pretraining.

4 An investigation of incoherent bases

Table 1 (last column) shows the mutual incoherence $\max_{t,c}|\langle \frac{\mathbf{pos}_t}{\|\mathbf{pos}_t\|}, \frac{\mathbf{ctx}_c}{\|\mathbf{ctx}_c\|}\rangle|$, as a measure of alignment between the positional basis and the context basis. The low incoherence in Table 1 (zero is impossible due to noise) means that the two bases are nearly orthogonal to each other. This weak alignment is a key structural requirement for sparse learning and is often associated with *restricted isometry* (Candes & Tao, 2005), *irrepresentable conditions* (Zhao & Yu, 2006), etc.

4.1 QK matrix decomposition: a study on induction heads

Induction heads (Elhage et al., 2021) are components in transformers that complete a sequence pattern based on observed past tokens, namely, predicting the next token $[B]$ based on observed

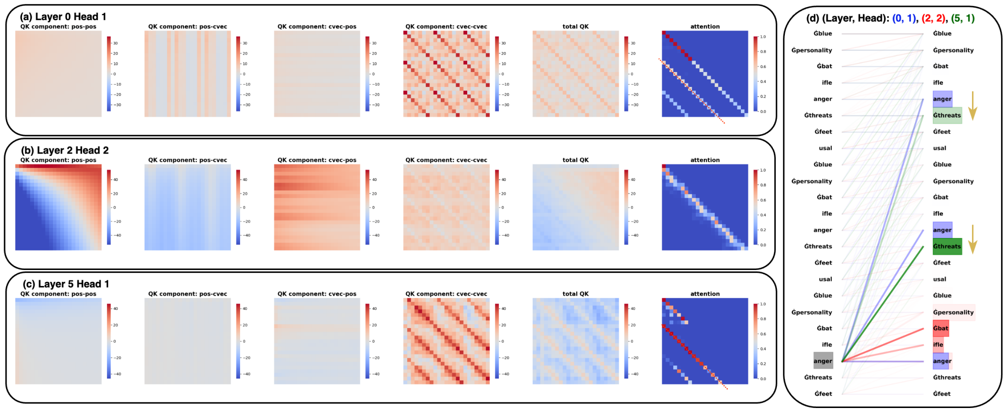

Figure 4: **Dissecting QK/attention**: GPT-2 on a repeated sequence of random tokens. **(a)(b)(c)**: we visualize $\mathbf{pos}$-$\mathbf{pos}$, $\mathbf{pos}$-$\mathbf{cvec}$, $\mathbf{cvec}$-$\mathbf{pos}$, $\mathbf{cvec}$-$\mathbf{cvec}$ QK components (first four columns of heatmaps), QK matrix (5-th column), and attention matrix (6-th column). **(d)**: We visualize the same three attention heads simultaneously, highlighting 'anger' token and its associated attention.

sequence $[A], [B], \ldots, [A]$. They are recently identified to explain in-context learning abilities of large language models. Surprisingly, induction heads even generalize on out-of-distribution data.

To dissect the self-attention mechanism, we decompose the QK matrices into components: assuming $\boldsymbol{\mu} = \mathbf{0}$, then for embedding vectors $\boldsymbol{h}, \boldsymbol{h}' \in \mathbb{R}^d$ we have

$$\begin{aligned} \boldsymbol{h}^\top \boldsymbol{W}^q (\boldsymbol{W}^k)^\top \boldsymbol{h} = \mathbf{pos}^\top \boldsymbol{W}^q (\boldsymbol{W}^k)^\top \mathbf{pos} + \mathbf{pos}^\top \boldsymbol{W}^q (\boldsymbol{W}^k)^\top \mathbf{cvec} \\ + \mathbf{cvec}^\top \boldsymbol{W}^q (\boldsymbol{W}^k)^\top \mathbf{pos} + \mathbf{cvec}^\top \boldsymbol{W}^q (\boldsymbol{W}^k)^\top \mathbf{cvec}\,. \end{aligned} \tag{7}$$

Each of the four components shows how much an attention head captures information from cross-pairs $\mathbf{pos}/\mathbf{cvec}$—$\mathbf{pos}/\mathbf{cvec}$ of an embedding. Although the global mean $\boldsymbol{\mu} \neq \mathbf{0}$ in reality, we find that it has little effect on interpretations.

Attention attribution for induction heads. Motivated by Equation 7, we decompose QK matrices into interpretable components that illuminate the mechanism of induction heads. We generate a sequence of 8 random tokens, repeat it twice, and then concatenate them into an input sequence for the pretrained GPT-2 model. For each layer/head, we calculate four QK matrix components; for example, for $\mathbf{pos}$-$\mathbf{pos}$ matrix, each entry is given by $\mathbf{pos}_i^\top \boldsymbol{W}^q (\boldsymbol{W}^k)^\top \mathbf{pos}_j$ where $i, j \leq T$.

Generally, we find that $\mathbf{pos}$-$\mathbf{pos}$ is smoothly dependent on positions, while $\mathbf{cvec}$-$\mathbf{cvec}$ exposes non-global effects of individual tokens. Depending on the magnitude of the four components, the "winning" component will determine the pattern of the attention matrix.

As shown in Figure 4, we identify three types of attention heads that are vital for induction heads.

- **Attention to self-tokens** (Layer 0, Head 1). The dominant component is $\mathbf{cvec}$-$\mathbf{cvec}$ (4-th heatmap in (a)), where visible diagonal lines indicate strong association between identical tokens, which translates to attention to previous identical tokens (blue lines in (d)).
- **Attention to neighboring tokens** (Layer 2, Head 2). The dominant component is $\mathbf{pos}$-$\mathbf{pos}$ (1-th heatmap in (b)), where upper right entries have higher values, resulting in attention to the previous few tokens (red lines in (d)), thanks to softmax and causal masking.
- **Attention to token being copied** (Layer 5, Head 1). The combination of the first two types gives rise to a new QK/attention pattern, where visible diagonal lines are *shifted* (yellow arrow) by one token; compare 6-th heatmaps in (a) & (c). Given token 'anger', as shown in grey in (d), attention heads look back for the next adjacent token, as shown in green.

Each type contains many representative heads including our handpicked ones; see Section D.1.

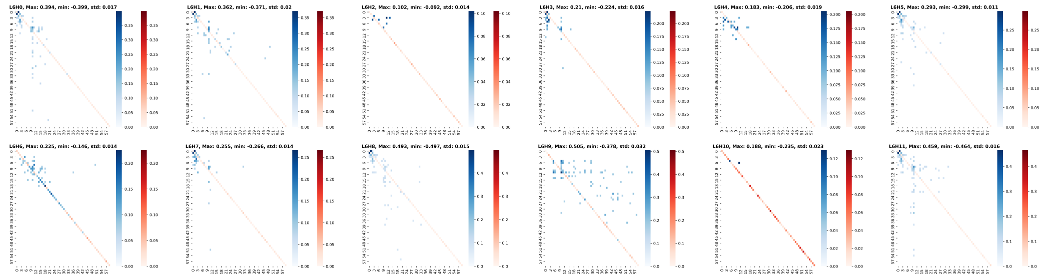

Figure 5: **Structure of attention weight matrices.** For any of the 12 attention heads (for layer $L = 6$ shown here) in GPT-2, we study the matrix $\boldsymbol{W} = \boldsymbol{W}^q(\boldsymbol{W}^k)^\top/\sqrt{d_{\text{head}}} \in \mathbb{R}^{d\times d}$. Red: we show the diagonal entries $\text{diagg}(\boldsymbol{W})$. Blue: we take off-diagonal matrix $\boldsymbol{W} - \text{diagg}(\boldsymbol{W})$ and rotate it by the right singular vectors of positional basis, then show the large absolute values.

4.2 Dissecting attention weight matrices

So far, we have observed that positional information is passed from earlier layers to later layers, yielding clear geometric structures. How does transformer layer $\texttt{TFLayer}_\ell$ enable this information flow? A natural hypothesis is that the weight matrix $\boldsymbol{W} := \boldsymbol{W}^q(\boldsymbol{W}^k)^\top/\sqrt{d_{\text{head}}}$ has a component that is aligned with the low-dimensional subspace where the positional basis lies. We empirically examine whether the following *low-rank plus noise* structure holds for certain heads.

$$\boldsymbol{W} = \boldsymbol{V}\boldsymbol{L}\boldsymbol{V}^\top + \boldsymbol{D} + \text{Noise} \tag{8}$$

where columns of $\boldsymbol{V} \in \mathbb{R}^{d\times K}$ are the top-K right singular vectors of positional basis matrix $\boldsymbol{P}$, $\boldsymbol{L} \in \mathbb{R}^{K\times K}$, and $\boldsymbol{D} \in \mathbb{R}^{d\times d}$ is a diagonal matrix.

In Figure 5, we take $\boldsymbol{D} = \text{diagg}(\boldsymbol{W})$ (shown in red), rotate the off-diagonal part of $\boldsymbol{W}$ by the right singular vectors of $\boldsymbol{P}$ and apply denoising, namely zeroing entries whose absolute values are smaller than a threshold. For many heads, the surviving large absolute values are concentrated in the top left ($K \approx 20$)—which suggests that indeed a significant component of $\boldsymbol{W}$ is aligned with the positional basis, supporting Equation 8.

4.3 Theoretical insight: kernel factorization

What are the desirable properties that incoherence structure induces in many trained transformers? It is well known in sparse coding and compressed sensing that incoherent basis facilitates recovery of sparse signals (Donoho & Stark, 1989; Donoho & Elad, 2003; Donoho, 2006; Candès et al., 2006).

Here we focus on the self-attention mechanism of transformers. By adopting the kernel perspective, we provide preliminary analysis for our following heuristics:

Incoherence enables a kernel to factorize into smaller components, each operating independently.

Given query/key matrices $\boldsymbol{W}^q, \boldsymbol{W}^k \in \mathbb{R}^{d\times d_{\text{head}}}$, we define the (asymmetric) kernel by

$$K_{\boldsymbol{W}}(\boldsymbol{z},\boldsymbol{z}') := \exp\left(\boldsymbol{z}^\top \boldsymbol{W}\boldsymbol{z}'\right) = \exp\left(\frac{\langle \boldsymbol{W}^q\boldsymbol{z}, \boldsymbol{W}^k\boldsymbol{z}'\rangle}{\sqrt{d_{\text{head}}}}\right), \qquad \text{recall } \boldsymbol{W} = \boldsymbol{W}^q(\boldsymbol{W}^k)^\top/\sqrt{d_{\text{head}}}.$$

Using $K_{\boldsymbol{W}}$, the attention can be expressed as kernel smoothing: for embeddings $(\boldsymbol{x}_t)_{t\le T} \subset \mathbb{R}^d$,

$$\text{AttnHead}(\boldsymbol{x}_t; K_{\boldsymbol{W}}) = \sum_{k\le t} \frac{K_{\boldsymbol{W}}(\boldsymbol{x}_k,\boldsymbol{x}_t)}{\sum_{k'\le t} K_{\boldsymbol{W}}(\boldsymbol{x}_{k'},\boldsymbol{x}_t)} v(\boldsymbol{x}_k) \tag{9}$$

where $v: \mathbb{R}^d \to \mathbb{R}$ is a generic value function. This kernel perspective is explored in Tsai et al. (2019), where it is argued that the efficacy of self-attention largely depends on the form of the kernel.

Suppose that there are two overcomplete bases $\mathcal{B}_1^0, \mathcal{B}_2^0 \subset \mathbb{R}^d$. For simplicity, assume that $\|\boldsymbol{u}\|_2 \le 1$ if $\boldsymbol{u} \in \mathcal{B}_1^0$ or $\mathcal{B}_2^0$. The mutual incoherence is $\text{incoh} := \max\left\{|\langle \boldsymbol{c},\boldsymbol{t}\rangle : \boldsymbol{c} \in \mathcal{B}_1^0, \boldsymbol{t} \in \mathcal{B}_2^0\right\}$. Consider the (extended) overcomplete basis $\mathcal{B}_\alpha := \{\lambda \boldsymbol{u} : \boldsymbol{u} \in \mathcal{B}_\alpha^0, \lambda \in [-1,1]\}$ where $\alpha \in \{1,2\}$. Given query/key vectors $\boldsymbol{x}^q, \boldsymbol{x}^k \in \mathbb{R}^d$, suppose that we can decompose them according to the two bases.

$$\boldsymbol{x}^q = \boldsymbol{c}^q + \boldsymbol{t}^q, \quad \boldsymbol{x}^k = \boldsymbol{c}^k + \boldsymbol{t}^k, \qquad \text{where } \boldsymbol{c}^q, \boldsymbol{c}^k \in \mathcal{B}_1;\ \boldsymbol{t}^q, \boldsymbol{t}^k \in \mathcal{B}_2. \tag{10}$$

We can generically decompose the kernel into a product of four components

$$K_{\boldsymbol{W}}(\boldsymbol{x}^q, \boldsymbol{x}^k) = K_{\boldsymbol{W}}(\boldsymbol{c}^q, \boldsymbol{c}^k) K_{\boldsymbol{W}}(\boldsymbol{c}^q, \boldsymbol{t}^k) K_{\boldsymbol{W}}(\boldsymbol{t}^q, \boldsymbol{c}^k) K_{\boldsymbol{W}}(\boldsymbol{t}^q, \boldsymbol{t}^k)\,.$$

Each kernel component measures cross similarity of pairs between $\boldsymbol{c}^q, \boldsymbol{t}^q$ and $\boldsymbol{c}^k, \boldsymbol{t}^k$, which then translates into a weight for the attention. Unfortunately, this general decomposition requires the individual kernels to share the same weight $\boldsymbol{W}$, which hinders capturing cross interactions flexibly.

It turns out that if the weight matrix is *sparsely represented* by the bases, then kernel flexibility can be achieved. To be precise, we will say that $\boldsymbol{W} \in \mathbb{R}^{d\times d}$ is s-sparsely represented by bases $\mathcal{B}, \mathcal{B}'$ if there exist $(a_k)_{k\le s} \subset [-1,1]$, $(\boldsymbol{u}_k)_{k\le s} \subset \mathcal{B}$, $(\boldsymbol{v}_k)_{k\le s} \subset \mathcal{B}'$ such that

$$\boldsymbol{W} = \sum_{k\le s} a_k \boldsymbol{u}_k \boldsymbol{v}_k^\top\,. \tag{11}$$

Theorem 2. *Let $\boldsymbol{W}_{11}, \boldsymbol{W}_{12}, \boldsymbol{W}_{21}, \boldsymbol{W}_{22} \in \mathbb{R}^{d\times d}$ be any matrices with the following properties: for $\alpha, \beta \in \{1,2\}$, $\boldsymbol{W}_{\alpha\beta} \in \mathbb{R}^{d\times d}$ is $O(1)$-sparsely represented by bases $\mathcal{B}_\alpha, \mathcal{B}_\beta$. Then for all $\boldsymbol{x}^q, \boldsymbol{x}^k \in \mathbb{R}^d$ satisfying Equation 10, $\boldsymbol{W} = \boldsymbol{W}_{11} + \boldsymbol{W}_{12} + \boldsymbol{W}_{21} + \boldsymbol{W}_{22}$ satisfies*

$$K_{\boldsymbol{W}}(\boldsymbol{x}^q, \boldsymbol{x}^k) = \big(1 + O(\mathrm{incoh})\big) \cdot K_{\boldsymbol{W}_{11}}(\boldsymbol{c}^q, \boldsymbol{c}^k) K_{\boldsymbol{W}_{12}}(\boldsymbol{c}^q, \boldsymbol{t}^k) K_{\boldsymbol{W}_{21}}(\boldsymbol{t}^q, \boldsymbol{c}^k) K_{\boldsymbol{W}_{22}}(\boldsymbol{t}^q, \boldsymbol{t}^k) \tag{12}$$

Moreover, Equation 12 holds with probability at least $1 - O((|\mathcal{B}_1^0| \cdot |\mathcal{B}_2^0|) \exp(-\mathrm{incoh}^2 \cdot d)$ if each $\boldsymbol{W}_{\alpha\beta}$ is replaced by $\boldsymbol{W}_{\alpha\beta} + \frac{\mathbf{Z}_{\alpha\beta}}{\sqrt{d}}$ where $(\mathbf{Z}_{\alpha\beta})_{kk'}$ is an independent subgaussian[1] random variable.

The factorization 12 says that each kernel component has a separate weight matrix, and all components contribute multiplicatively to $K_{\boldsymbol{W}}$. The "moreover" part generalizes the sparse representation notion by allowing additive noise, which matches the empirical structure in Equation 8.

Remark 1. *If we suppose* incoh $\asymp d^{-\gamma}$ *with* $1/2 > \gamma > 0$, *then the high probability statement is nontrivial if* $|\mathcal{B}_1^0| \cdot |\mathcal{B}_2^0| = o(\exp(d^{1-2\gamma}))$. *This dictionary size limit is generally reasonable.*

5 Why training format matters: smoothness perspective

5.1 Training with token randomization

We train transformers on three different training data: (i) baseline—sequences of length $T = 512$ sampled from the first 10K samples of OpenWebText, (ii) partial randomization—sequences of the same data source but the latter half is replaced by random tokens uniformly sampled in the vocabulary, (iii) full randomization—input sequences are fully random tokens.

Figure 6 (a)–(c) show the positional bases of two selected layers (L1 & L5) from the three settings. As darker colors represent later positions, we find that randomization destroys geometric structures of positional basis at exact positions where randomization takes place.

5.2 Simple experiments on addition tasks

We explore a simple arithmetic task—*Addition*, where inputs are formatted as a string "$a + b = c$" with a, b, c represented by digits of a certain length. We sample the length of each addition component uniformly from $\{L/2, \ldots, L\}$ where $L = 10$ and then (i) in the "carry" setting, sample digits independently, and (ii) in the "no-carry" setting, examples involving carry are removed. For both settings, the output order is reversed (Lee et al., 2023). Training transformers on datasets under the two settings, we discover similar phenomena, so we only present results for (i).

In Figure 6 (d)–(e), we visualize two exemplary attention heads in a way similar to Figure 4. We find many fractured or discontinuous QK matrices and their $\mathbf{pos}$-$\mathbf{pos}$ components. Likely associated with this discontinuity pattern, we find the transformer has difficulty generalizing to longer or shorter sequences (failure of length generalization). See Section E for details.

[1]We say that a random variable ξ is subgaussian if $\mathbb{E}[\xi] = 0$ and $\mathbb{E}[\exp(\lambda\xi)] \le \exp(\lambda^2/2)$ for all $\lambda \in \mathbb{R}$.

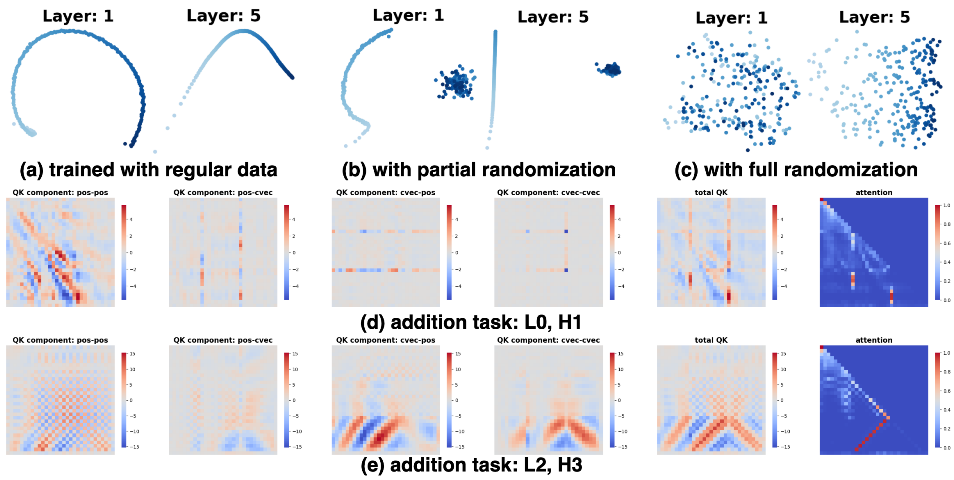

Figure 6: **(a)(b)(c): Token randomization experiments** (first row). Positional basis of transformers trained on unmodified text sequences, sequences with the latter half replaced by random tokens, purely random tokens, respectively. **(d)(e): Addition experiment** (second/third row). *Fractured/discontinuous* patterns likely cause length generalization to fail.

6 Related work

Analyses of transformers have attracted research interest since (Vaswani et al., 2017). Many studies on GPT-2 (Radford et al., 2019) and BERT (Devlin et al., 2018) show that last-layer contextualized embeddings capture linguistic structure and exhibit excellent downstream performance (Hewitt & Manning, 2019; Chi et al., 2020; Thompson & Mimno, 2020). Fewer papers focus on the geometry or intermediate-layer embeddings: in Ethayarajh (2019), it is found that later-layer embeddings are increasingly anisotropic and context-specific; Cai et al. (2020); Reif et al. (2019); Hernandez & Andreas (2021); Gao et al. (2019) observed interesting geometric structures and artifacts without thorough analysis; Yeh et al. (2023) provide visualization tools for embeddings. Our decomposition reveals consistent geometry and explains observed artifacts (anisotropic, spiral shape, etc.).

Many variants of positional embedding are proposed (Shaw et al., 2018; Dai et al., 2019; Su et al., 2021; Scao et al., 2022; Press et al., 2021) since Vaswani et al. (2017). Since GPT-4, many papers focus on length generalization for arithmetic tasks (Kazemnejad et al., 2023; Lee et al., 2023). Prior analyses on positional embeddings focus only on static (0-th layer) embeddings for selected transformers (Wang et al., 2020; Ke et al., 2020; Wang & Chen, 2020; Tsai et al., 2019), whereas we provide a complete picture.

Prior work on LSTMs finds decomposition-based methods can enhance interpretability (Murdoch et al., 2018). Understanding the inner workings of transformers is usually done through visualizing the attention heads (Clark et al., 2019; Wang et al., 2022). The emergence of induction head Elhage et al. (2021); Olsson et al. (2022) is supported by attention visualization, which is further reinforced by our analysis.

7 Limitations

In this paper, we mostly focus on pretrained transformers due to limited computational resources. It would be interesting to investigate the impact of input/prompt formats on the geometry of embeddings over the course of training, especially for different linguistic tasks and arithmetic tasks.

Also, we mostly focus on the mean vectors $\mathbf{pos}_t$ and $\mathbf{ctx}_c$ but not study $\mathbf{resid}_{c,t}$ thoroughly. It would be interesting to study the higher-order interaction in $\mathbf{resid}_{c,t}$ and propose a nonlinear decomposition of embeddings, which is left to future work.

REFERENCES

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

S Allen Broughton and Kurt Bryan. *Discrete Fourier analysis and wavelets: applications to signal and image processing*. John Wiley & Sons, 2018.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Xingyu Cai, Jiaji Huang, Yuchen Bian, and Kenneth Church. Isotropy in the contextual embedding space: Clusters and manifolds. In *International Conference on Learning Representations*, 2020.

Emmanuel J Candes and Terence Tao. Decoding by linear programming. *IEEE transactions on information theory*, 51(12):4203–4215, 2005.

Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on information theory*, 52(2):489–509, 2006.

Ethan A Chi, John Hewitt, and Christopher D Manning. Finding universal grammatical relations in multilingual bert. *arXiv preprint arXiv:2005.04511*, 2020.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*, 2019.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

David Donoho, Matan Gavish, and Elad Romanov. Screenot: Exact mse-optimal singular value thresholding in correlated noise. *The Annals of Statistics*, 51(1):122–148, 2023.

David L Donoho. Compressed sensing. *IEEE Transactions on information theory*, 52(4):1289–1306, 2006.

David L Donoho and Michael Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via ℓ^1 minimization. *Proceedings of the National Academy of Sciences*, 100(5):2197–2202, 2003.

David L Donoho and Philip B Stark. Uncertainty principles and signal recovery. *SIAM Journal on Applied Mathematics*, 49(3):906–931, 1989.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. *arXiv preprint arXiv:1909.00512*, 2019.

Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. Representation degeneration problem in training natural language generation models. *arXiv preprint arXiv:1907.12009*, 2019.

Evan Hernandez and Jacob Andreas. The low-dimensional linear geometry of contextualized word representations. *arXiv preprint arXiv:2105.07109*, 2021.

John Hewitt and Christopher D Manning. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4129–4138, 2019.

Amirhossein Kazemnejad, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Payel Das, and Siva Reddy. The impact of positional encoding on length generalization in transformers. *arXiv preprint arXiv:2305.19466*, 2023.

Guolin Ke, Di He, and Tie-Yan Liu. Rethinking positional encoding in language pre-training. *arXiv preprint arXiv:2006.15595*, 2020.

Nayoung Lee, Kartik Sreenivasan, Jason D Lee, Kangwook Lee, and Dimitris Papailiopoulos. Teaching arithmetic to small transformers. *arXiv preprint arXiv:2307.03381*, 2023.

W James Murdoch, Peter J Liu, and Bin Yu. Beyond word importance: Contextual decomposition to extract interactions from lstms. *arXiv preprint arXiv:1801.05453*, 2018.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

Mark A Pinsky. *Introduction to Fourier analysis and wavelets*, volume 102. American Mathematical Soc., 2008.

Ofir Press, Noah A Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. *arXiv preprint arXiv:2108.12409*, 2021.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B Viegas, Andy Coenen, Adam Pearce, and Been Kim. Visualizing and measuring the geometry of bert. *Advances in Neural Information Processing Systems*, 32, 2019.

Mark Rudelson and Roman Vershynin. Sampling from large matrices: An approach through geometric functional analysis. *Journal of the ACM (JACM)*, 54(4):21–es, 2007.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. Self-attention with relative position representations. *arXiv preprint arXiv:1803.02155*, 2018.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021.

Laure Thompson and David Mimno. Topic modeling with contextualized word representation clusters. *arXiv preprint arXiv:2010.12626*, 2020.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Yao-Hung Hubert Tsai, Shaojie Bai, Makoto Yamada, Louis-Philippe Morency, and Ruslan Salakhutdinov. Transformer dissection: a unified understanding of transformer's attention via the lens of kernel. *arXiv preprint arXiv:1908.11775*, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Jesse Vig and Yonatan Belinkov. Analyzing the structure of attention in a transformer language model. *arXiv preprint arXiv:1906.04284*, 2019.

Benyou Wang, Lifeng Shang, Christina Lioma, Xin Jiang, Hao Yang, Qun Liu, and Jakob Grue Simonsen. On position embeddings in bert. In *International Conference on Learning Representations*, 2020.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *arXiv preprint arXiv:2211.00593*, 2022.

Yu-An Wang and Yun-Nung Chen. What do position embeddings learn? an empirical study of pre-trained language model positional encoding. *arXiv preprint arXiv:2010.04903*, 2020.

Catherine Yeh, Yida Chen, Aoyu Wu, Cynthia Chen, Fernanda Viégas, and Martin Wattenberg. Attentionviz: A global view of transformer attention. *arXiv preprint arXiv:2305.03210*, 2023.

Peng Zhao and Bin Yu. On model selection consistency of lasso. *The Journal of Machine Learning Research*, 7:2541–2563, 2006.