Learned Transformer Position Embeddings Have a Low-Dimensional Structure

Ulme Wennberg
Division of Speech, Music and Hearing
KTH Royal Institute of Technology
ulme@kth.se

Gustav Eje Henter
Division of Speech, Music and Hearing
KTH Royal Institute of Technology
ghe@kth.se

Abstract

Position embeddings have long been essential for sequence-order encoding in transformer models, yet their structure is underexplored. This study uses principal component analysis (PCA) to quantitatively compare the dimensionality of absolute position and word embeddings in BERT and ALBERT. We find that, unlike word embeddings, position embeddings occupy a low-dimensional subspace, typically utilizing under 10% of the dimensions available. Additionally, the principal vectors are dominated by a few low-frequency rotational components, a structure arising independently across models.

1 Introduction

Transformers, as introduced by Vaswani et al. (2017), have significantly advanced the field of natural language processing, excelling in tasks like machine translation (Lample et al., 2018), question answering (Yamada et al., 2020), information extraction (Wadden et al., 2019; Lin et al., 2020), and text generation (Radford et al., 2018; Brown et al., 2020). The ability to encode positional information is vital in these models, since the transformer architecture otherwise does not take order into account.

Despite their widespread use, the structure of absolute position embeddings in NLP models like BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2020), and ELECTRA (Clark et al., 2020), as well as vision models like the vision transformer (Dosovitskiy et al., 2021) and BEIT (Bao et al., 2022), remains underexplored. Our research aims to address this gap.

This paper investigates the structure of learned absolute position embeddings in greater detail than before. Specifically, we apply principal components analysis to the learned position embeddings across 12 different transformer-based language models. This yields several novel observations:

- Unlike word embeddings, position embeddings occupy a low-dimensional subspace.
- Variation within this subspace takes the shape of mutually orthogonal periodic components operating pairwise at different frequencies.

These trends are consistent across different models. Our findings resemble mechanisms for mathematical processing recently observed in transformers and suggest new ways in which sequence order can be encoded and learned in transformer models.

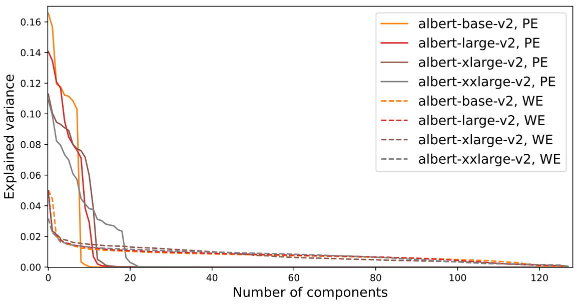


Figure 1: Variance explained by each individual principal component for position (PE, solid lines) and word (WE, dashed) embeddings across four ALBERT models. Each component explains less variance than the previous one by definition. Unlike word embeddings, position embeddings occupy a low-dimensional subspace.

2 Background

Transformer-based language models, such as those by Vaswani et al. (2017), have dramatically changed natural language processing by effectively integrating information across long distances in a sequence. Central to the functionality of these models are position embeddings, which enable the encoding of sequence order—an essential aspect in otherwise order-agnostic transformer architectures.

Content embeddings z_i in transformers are constructed as $z_i = e_W(x_i) + e_P(i)$, where x_i is the token at position i, e_W represents word embeddings, and $e_P(i)$ is the position embedding vector of position i; E_P will denote the matrix obtained by stacking all row-vectors e_P. This setup allows the final representation of tokens in a sequence

Proceedings of the 9th Workshop on Representation Learning for NLP (RepL4NLP-2024), pages 237–244
August 15, 2024 ©2024 Association for Computational Linguistics

Table 1: Variance in principal components (PCs) of word and position embeddings for twelve different language models. PCA of sinusoidal position embeddings (Vaswani et al., 2017) is also included for reference.

(a) Position embeddings

Model	Tot. PCs	Top 3	Top 5	Top 10	$N_{50\%}$	(%)
Sinusoidal	128	0.28	0.37	0.52	10	7.8%
albert-base-v1	128	0.50	0.69	0.99	4	3.1%
albert-base-v2	128	0.44	0.67	1.00	4	3.1%
albert-large-v1	128	0.45	0.63	0.95	4	3.1%
albert-large-v2	128	0.40	0.61	0.96	4	3.1%
albert-xlarge-v1	128	0.34	0.51	0.88	5	3.9%
albert-xlarge-v2	128	0.31	0.49	0.89	6	4.7%
albert-xxlarge-v1	128	0.27	0.41	0.68	7	5.5%
albert-xxlarge-v2	128	0.29	0.44	0.72	6	4.7%
bert-base-uncased	512	0.25	0.38	0.62	8	1.6%
bert-base-cased	512	0.28	0.42	0.64	7	1.4%
bert-large-uncased	512	0.23	0.33	0.53	10	2.0%
bert-large-cased	512	0.27	0.41	0.65	7	1.4%

(b) Word embeddings

Model	Tot. PCs	Top 3	Top 5	Top 10	$N_{50\%}$	(%)
albert-base-v1	128	0.07	0.11	0.19	37	28.9%
albert-base-v2	128	0.11	0.15	0.21	39	30.5%
albert-large-v1	128	0.08	0.12	0.20	34	26.6%
albert-large-v2	128	0.09	0.13	0.20	39	30.5%
albert-xlarge-v1	128	0.09	0.13	0.23	27	21.1%
albert-xlarge-v2	128	0.09	0.13	0.21	33	25.8%
albert-xxlarge-v1	128	0.08	0.11	0.18	39	30.5%
albert-xxlarge-v2	128	0.07	0.11	0.18	39	30.5%
bert-base-uncased	768	0.09	0.10	0.12	185	24.1%
bert-base-cased	768	0.05	0.07	0.10	164	21.4%
bert-large-uncased	1024	0.07	0.08	0.10	238	23.2%
bert-large-cased	1024	0.07	0.08	0.11	198	19.3%

to depend on token positions, which is crucial to adequately model contextual effects in text.

While Vaswani et al. (2017) used a fixed, non-learnable encoding scheme for position embeddings, subsequent work has aimed to enhance the expressiveness and efficiency of position encoding. BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) used learnable, data-driven position embeddings to better capture positional dependencies. ALBERT (Lan et al., 2020) refined this approach by introducing factorized embedding parameterizations, reducing model size and complexity while maintaining performance. Research has shown that varying word embedding sizes based on frequency can significantly improve computational efficiency and performance (Grave et al., 2017; Baevski and Auli, 2019; Dai et al., 2019), but varying sizes between word and position embeddings has not been explored, with standard practice being to use the same dimensionality for both.

Most work on position embeddings examines their impact on model performance, not their intrinsic properties. Existing results show that self-attention tends to localize in models using absolute position embeddings (Clark et al., 2019; Htut et al., 2019) and that these embeddings have translation-equivariant structure (Wennberg and Henter, 2021).

3 Dimensionality Analysis of Embeddings

We now analyze the dimensionality of word and position embeddings in transformer-based language models. A deeper dive into into the structure of position embeddings is reserved for Sec. 4.

To understand the structural characteristics of position and word embeddings, we extracted both embedding types from various pre-trained transformer models, specifically twelve different versions of ALBERT and BERT provided by Hugging Face (Wolf et al., 2020). We then applied principal component analysis (PCA) to these embeddings, analyzing each type separately. PCA computes a linear transformation that decomposes high-dimensional data into orthogonal vectors representing the primary axes of variation. Dimensionality reduction is performed by keeping only the k leading principal components (PCs).

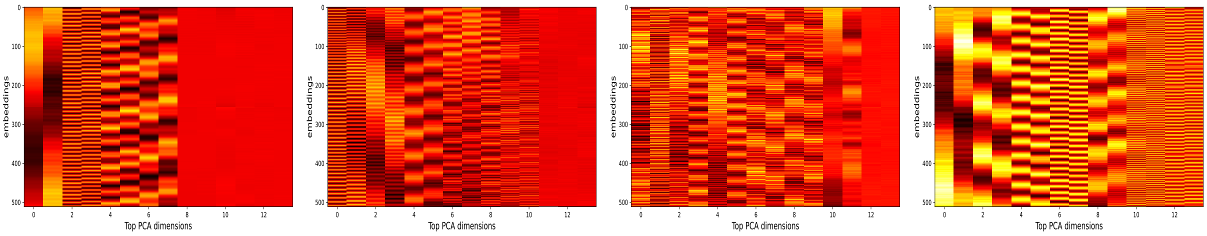
Table 1 reports on the results of the PCA analysis. Specifically, it shows how much of the total variance among embedding vectors of each same type that can be explained by the top 3, 5, and 10 principal components, as well as how many components are needed to explain at least 50% of the variation between the vectors (denoted $N_{50\%}$). This allows us to assess and compare the effective dimensionality between position and word embeddings.

From the tables, we see that position embeddings have a significantly lower-dimensional structure compared to word embeddings, suggesting that positional information is encoded more compactly. All ALBERT and BERT models considered have 50% of their variance in the first 1.4–5.5% of the principal components or less, while word embeddings require 19–30% of the PCs to achieve the same result. Figure 1 graphs the variance explained by individual principal components in detail for the ALBERT v2 models, finding that position-embedding vectors lie almost perfectly on a subspace of 10 to 20 dimensions, whereas word embeddings use the entire 128-dimensional space.

4 Analyzing the Principal Components

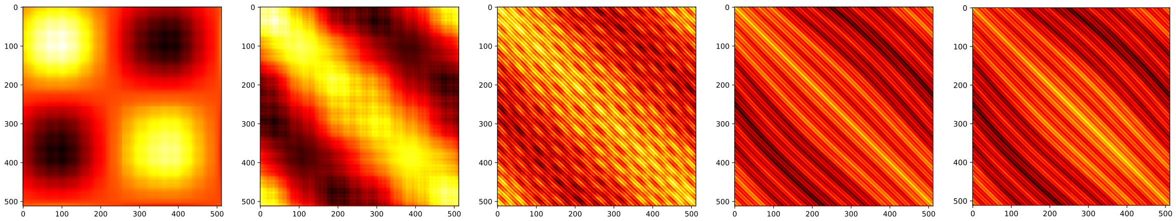
Having established the low dimensionality of position embeddings, we next explore what the uncovered principal components represent and how they contribute to the embedding structure.

First, we plot (in Figure 2) the leading principal



(a) albert-base-v2 (b) albert-large-v2 (c) albert-xlarge-v2 (d) albert-xxlarge-v2

Figure 2: Heatmaps visualizing the top 14 PCs of the position-embedding matrices E_P of the ALBERT v2 models. Best viewed in Adobe Acrobat to avoid blurry rendering. The full matrix with all PCs can be found in Figure 7.



(a) 1 component (b) 2 components (c) 6 components (d) 10 components (e) All components

Figure 3: Heatmaps visualizing the matrix $P = E_P E_P^T$ of position-embedding inner products in albert-base-v2, when E_P is approximated by its top k PCs. The greater the value of the inner product, the lighter the color.

components of a few models to visually interpret the dominant patterns in the position embeddings. This reveals intriguing patterns. In all cases, the first ten components take the form of smooth, periodic oscillations as a function of position, indicative of simple harmonic structure. Although the specific ordering and frequencies change between models, components come in pairs that exhibit similar periodic structure, like sine and cosine representing cyclical motion. For all models except the largest, the highest components plotted appear flat and uniform red (i.e., close to zero), reflecting the limited dimensionality of the position embeddings.

Figure 5 in the appendix demonstrates, through Fourier analysis, that the sequence of principal component scores contains only a few dominant frequencies, which accounts for their periodic appearance. The peak frequencies observed, such as those representing 1, 5, 15, and 49 revolutions as i runs through its full range from 0 to 511 in the case of albert-base-v2, are relatively low. This finding is distinct from the sinusoidal position embeddings described by Vaswani et al. (2017), which utilize 512 sinusoids of equal magnitude. Unlike the principal component scores, these sinusoids are not mutually orthogonal and are designed with different objectives for encoding position in a sequence.

By computing the matrix $E_P E_P^T$, which contains the inner products between all pairs of position embeddings, it has been found that learned position embeddings tend to exhibit translation equivariance (Wennberg and Henter, 2021). In contrast, classic sinusoidal position embeddings display weak inner products between off-diagonal elements, suggesting an absence of such patterns (Wang and Chen, 2020). By repeating this inner-product experiment, but approximating the position-embedding matrix E_P by its top k principal components, we can see how translation-equivariant structure (where each row of the matrix is a translation of the one above it) is rapidly created using only a few principal components for the albert-v2-base model in Figure 3.

Finally, as PCA is a dimensionality-reduction technique, we can visualize all 512 albert-base-v2 position-embedding vectors in two dimensions by means of a scatter plot of their two leading principal components, as shown in Figure 4. We observe a very clear rotational structure, where as the position i goes from 0 to 511, the 2D representation of $e_P(i)$ almost completes a full clockwise turn. Other principal-component pairs show similar patterns, but complete multiple rotations as i runs through the full range of position indices.

Figure 4 exhibits two outliers from the circular pattern, namely vectors 0 and 511 (the first and last). This is likely due to how the model is trained: position embeddings are only ever used after being summed with a word embedding, and the the first and the last sequence positions are always assigned the specific tokens “CLS” and “SEP”, respectively.

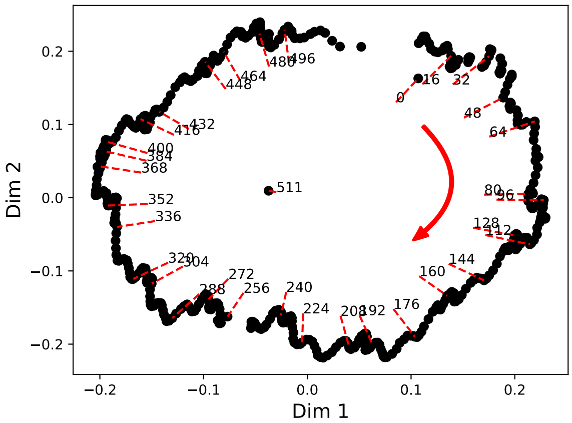


Figure 4: Scatter plot of albert-base-v2 position embeddings reduced to two dimensions using PCA. For clarity, every 16th position is annotated.

This means that, unlike at all other positions, these two vectors are largely arbitrary, e.g., adding any vector v to $e_P(0)$ while subtracting the same vector from $e_W(\text{CLS})$ leads to the exact same z_0.

5 Discussion and Implications

We have demonstrated that position embeddings operate within a significantly lower-dimensional space compared to word embeddings. This likely reflects their role in encoding less complex, but nonetheless essential structural information.

Opportunities for Transformer Models: Intuitively, our findings present an opportunity to streamline embeddings and reduce computational demands without compromising the model's ability to interpret linguistic contexts. For example, one could utilize factorized embedding parametrizations of different dimensionalities for position embeddings versus word embeddings, similar to how ALBERT mentions the possibility to use different embedding dimensionalities for different word tokens (Lan et al., 2020), although they opted not to do so. To further refine model inductive biases based on the patterns we observed, learned position embeddings could be initialized or otherwise incentivized to have rotational structure, e.g., being parameterized by sines and cosines with a learnable frequency. This would differ from the rotational position embeddings (RoPE) of by Su et al. (2024), whose rotational frequencies are not learnable.

We observed a consistently low-dimensional structure of position embeddings among a wide class of transformer models. This supports the soundness of the heuristic approach for creating position embeddings used in Longformer (Beltagy et al., 2020) – where pre-trained RoBERTa position embeddings were used as a starting point for training a new model – and further suggests that re-using learned position embeddings from older models may be useful as a general strategy.

Insights into Embedding Ordered Sequences: Transformers, particularly those like ALBERT models which have undergone extensive training, exhibit an intriguing pattern in their position embeddings. These models often utilize approximately 10 principal components—closely aligning with $2^9 = 512$, the typical maximum sequence length. This choice of dimensionality suggests that each dimension may function akin to a binary system, with each principal component potentially implementing a sine or cosine curve. Such a structure effectively splits the data, allowing for a compact yet robust representation of sequence positions.

This method of embedding sequences as concurrent rotations in low-dimensional spaces indicates a standardized approach to processing sequential data via embeddings. This geometric encoding strategy is echoed in findings across several recent studies. Nanda et al. (2023) noted that transformer models trained on mathematical tasks often use a "clock algorithm" in their latent spaces, enabling modular arithmetic. Similarly, Zhong et al. (2023) and Wennberg and Henter (2024) observed analogous rotational patterns in numerical embeddings, whether trained from scratch on mathematical tasks or using language-modeling techniques.

These observations highlight the potential of using geometric transformations as a unified method to encode sequential information across diverse applications, like time-series analysis, where precision and optimized data representation are crucial.

6 Conclusions and Future Work

We have found that learned position embeddings in a range of transformer language models differ from the behavior of word embeddings, in that position embeddings are confined to a low-dimensional linear subspace. We furthermore find evidence that this subspace takes the form of a few orthogonal rotational components at different frequencies.

Interesting future directions to explore include studying position embeddings in other domains, such as vision, and leveraging our findings to devise more efficient transformer variants with improved inductive biases for modeling sequence data.

Limitations

This study examined a select number of transformer models, using principal component analysis. PCA only considers linear subspaces for dimensionality reduction. Consequently, our analysis can only be interpreted as an upper-bound estimate of the intrinsic dimensionality of the manifold of which position embeddings reside, and may overlook nonlinear relationships within the embeddings. In other words, the actual dimensionality of the position-embedding manifold may be lower than our estimates, if it is nonlinear.

Our analysis is limited to a set of twelve different transformer-based language models that use learned absolute position embeddings. With our focus on absolute position embeddings, we did not study alternative position embeddings such as RoPE (Su et al., 2024). Although including additional position-embedding schemes would indeed be interesting, adapting our analysis methodology to RoPE is not straightforward, since it implements positional dependence differently, and in particular not by summing word embeddings with explicit postion embedding vectors $e_P(i)$. Additionally, it should be said that even though many recent language models utilize RoPE, models in other domains such as computer vision (Dosovitskiy et al., 2021; Bao et al., 2022) still use absolute position embeddings like the ones analyzed in this paper.

Furthermore, our investigation is confined to the input embeddings z_i of the models we study. This means that we cannot tell how the structure and dimensionality of these vectors may change during processing, as they pass through successive internal layers of the models and become increasingly context-dependent.

Finally, our study does not specifically analyze how the low-dimensional manifolds we uncovered influence the transformer self-attention. Investigating this might shed light on why these low-dimensional manifolds emerge in the first place.

Ethics Statement

To the best of our knowledge, this paper, which focuses on the analysis of learned position embeddings in transformer models, does not directly raise any ethical concerns.

References

Alexei Baevski and Michael Auli. 2019. Adaptive input representations for neural language modeling. In *Proc. ICLR*.

Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. 2022. BEiT: BERT pre-training of image transformers. In *Proc. ICLR*.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *Preprint*, arXiv:2004.05150.

T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, and et al. 2020. Language models are few-shot learners. In *Proc. NeurIPS*, pages 1877–1901.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? An analysis of BERT's attention. In *Proc. BlackboxNLP@ACL*, pages 276–286.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Pre-training transformers as energy-based cloze models. In *Proc. EMNLP*, pages 285–294.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proc. ACL*, pages 2978–2988.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. NAACL-HLT*, pages 4171–4186.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. ICLR*.

Edouard Grave, Armand Joulin, Moustapha Cissé, David Grangier, and Hervé Jégou. 2017. Efficient softmax approximation for GPUs. In *Proc. ICML*, pages 1302–1310.

Phu Mon Htut, Jason Phang, Shikha Bordia, and Samuel R. Bowman. 2019. Do attention heads in BERT track syntactic dependencies? *Preprint*, arXiv:1911.12246.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Phrase-based & neural unsupervised machine translation. In *Proc. EMNLP*, pages 5039–5049.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *Proc. ICLR*.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proc. ACL*, pages 7999–8009.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *Preprint*, arXiv:1907.11692.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. 2023. Progress measures for grokking via mechanistic interpretability. In *Proc. ICLR*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. `https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf`. Accessed: 2024-05-16.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. 2024. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. NeurIPS*, pages 5998–6008.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *Proc. EMNLP-IJCNLP*, pages 5784–5789.

Yu-An Wang and Yun-Nung Chen. 2020. What do position embeddings learn? An empirical study of pre-trained language model positional encoding. In *Proc. EMNLP*, pages 6840–6849.

Ulme Wennberg and Gustav Eje Henter. 2021. The case for translation-invariant self-attention in transformer-based language models. In *Proc. ACL-IJCNLP*, pages 130–140.

Ulme Wennberg and Gustav Eje Henter. 2024. Exploring internal numeracy in language models: A case study on ALBERT. *Preprint*, arXiv:2404.16574.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proc. EMNLP System Demonstrations*, pages 38–45.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proc. EMNLP*, pages 6442–6454.

Ziqian Zhong, Ziming Liu, Max Tegmark, and Jacob Andreas. 2023. The clock and the pizza: Two stories in mechanistic explanation of neural networks. In *Proc. NeurIPS*, pages 27223–27250.

A Appendix

For the interested reader, this appendix provides supplementary visual data to complement the analyses discussed in the main sections of the paper.

Figure 5 presents the summed frequency magnitude spectrum of the sequence of the principal component scores, based on the position embeddings from ALBERT base-v2, emphasizing dominant frequencies with a normalized Nyquist limit of 0.5 and a logarithmic magnitude scale.

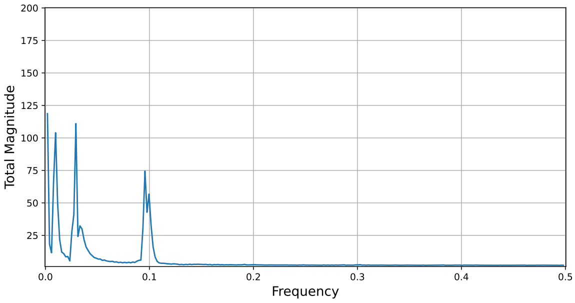


Figure 5: Frequency magnitude spectrum of principal component scores from albert-base-v2. The plot displays the sum of Fourier magnitudes across all the principal components, thus highlighting dominant frequencies. Frequencies are normalized with 0.5 as the Nyquist limit, and the plot uses a logarithmic y-axis.

Furthermore, Figure 6 depicts the frequency magnitude spectrum for each of the top 10 principal components in the sequence of the principal component scores, highlighting the unique spectral contributions of each principal component.

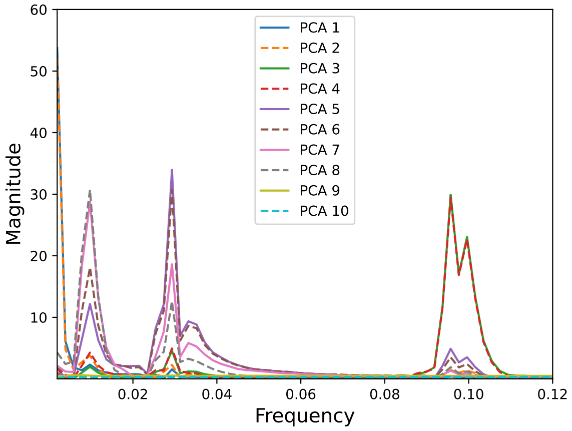
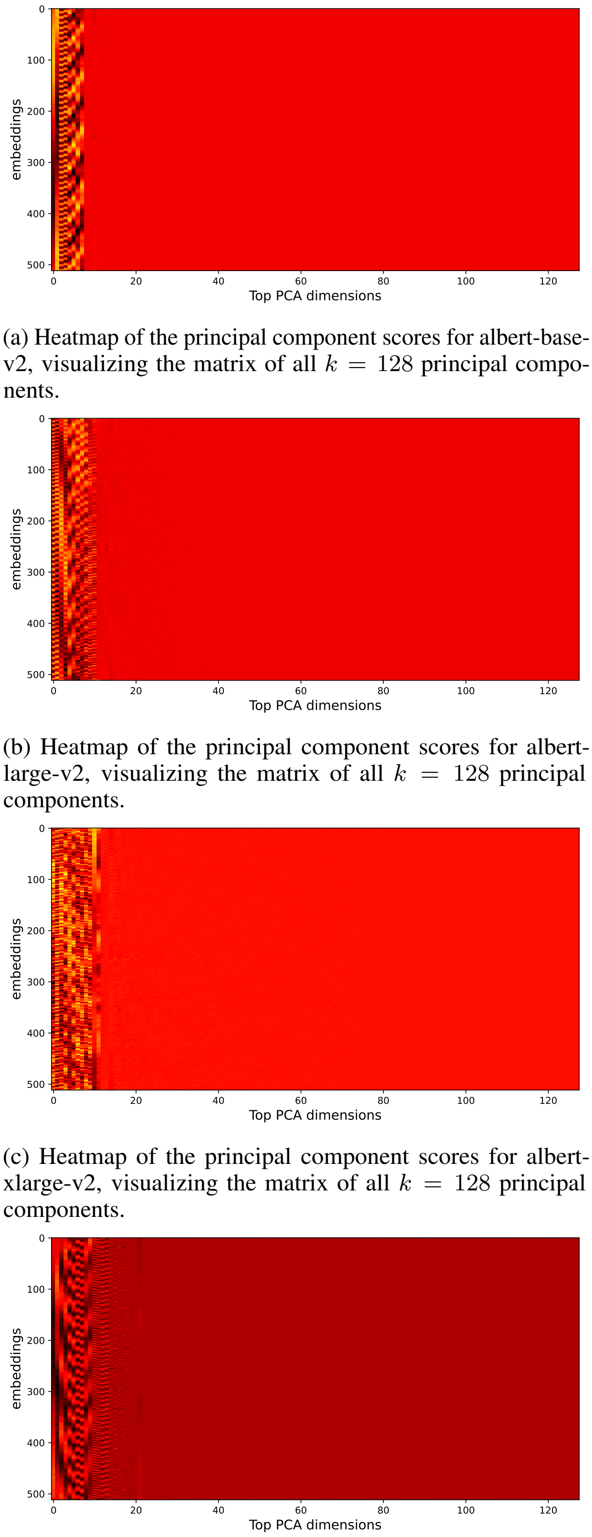


Figure 6: Frequency magnitude spectrum for each of the top 10 principal component scores from albert-base-v2. The plot displays the Fourier magnitudes for each principal component, thus highlighting their individual contributions. Frequencies are normalized with 0.5 as the Nyquist limit.

Figure 7 presents extended heatmaps representing the entirety of the principal component scores analyzed for various ALBERT models. It is easily noticeable that only the leftmost principal components contribute meaningfully to the variability in the data.



(a) Heatmap of the principal component scores for albert-base-v2, visualizing the matrix of all $k = 128$ principal components.

(b) Heatmap of the principal component scores for albert-large-v2, visualizing the matrix of all $k = 128$ principal components.

(c) Heatmap of the principal component scores for albert-xlarge-v2, visualizing the matrix of all $k = 128$ principal components.

(d) Heatmap of the principal component scores for albert-xxlarge-v2, visualizing the matrix of all $k = 128$ principal components.

Figure 7: Extended heatmaps representing the entirety of the principal component scores analyzed for various ALBERT models, highlighting the contribution of the leftmost components to the variability in the data.