LOST OR LIBERATED? A DIVE INTO BIDIRECTIONAL TRANSFORMER LMS WITHOUT POSITIONAL ENCODING

Urchade Zaratiana
zaratiana@lipn.fr
Laboratoire Informatique de Paris Nord

ABSTRACT

Recent studies have shown that Autoregressive Transformer Language Models (LMs) can generate text sequences without relying on positional encodings (PEs). This capability is attributed to the causal masks in these models, which prevent tokens from accessing information from future tokens, allowing implicit learning of token positions. On the other hand, Bidirectional LMs, such as BERT, tend to underperform on masked language modeling tasks when PEs are omitted. This performance dip arises because transformer layers are inherently permutation equivariant; without PEs or masks, they cannot differentiate token positions, making bidirectional processing difficult. In this analytical study, we examine a variant of bidirectional Transformer LM that operates without PEs but incorporates causal masks in its initial layers. Our findings reveal that this configuration yields performance metrics on masked language modeling tasks that are on par with traditional transformers that use PEs. However, when tested on the GLUE language understanding benchmark, the model without PEs exhibits diminished performance. These results highlight the importance of positional encodings in bidirectional LMs and indicate that pretraining loss might not always correlate with performance on downstream tasks.

1 INTRODUCTION

Transformers, as initially introduced by Vaswani et al. (2023), have propelled groundbreaking advancements across a multitude of application domains. One of the most important components of the transformer architecture is the positional encoding, designed to counteract the model's inherent permutation equivariance. In other words, without positional encodings, transformers exhibit a behavior where if the input sequence is permuted, the output will similarly be permuted but retain the same values. To address this, the most common approach is to inject positional encodings into the input sequence. Absolute positional encoding (**APE**) Gehring et al. (2017), for instance, creates a unique vector for each position ID and then adds this vector to the corresponding input token embedding. However, a limitation arises with absolute PE: during training, it learns embeddings for a fixed number of positions, often up to a predefined limit like 512 for models such as BERT (Devlin et al., 2019). This means that, post-training, it cannot be directly applied to sequences longer than this predefined limit. To circumvent this limitation, relative positional encodings (**RelPE**) (Raffel et al., 2020; Su et al., 2021) have been introduced. These do not tie positional information to fixed positions but rather learn embeddings that represent the relative distances or relationships between pairs of tokens in a sequence. Some popular RelPE includes *T5 relative PE* (Raffel et al., 2020), *Rotary embeddings* (Su et al., 2021), and *ALiBi* (Press et al., 2022).

Recent research has highlighted the feasibility of training autoregressive transformer language models without the use of positional embeddings (Irie et al., 2019; Haviv et al., 2022). This capability is believed to be influenced by the model's integration of causal masks, allowing them to discern token positions within sequences. In a recent study by Kazemnejad et al. (2023), a theoretical framework was presented, suggesting that a causal Transformer Language Model (TLM) can effectively emulate both absolute and relative positional encodings. Interestingly, their findings indicated that autoregressive models trained without positional encodings, but only with causal masks, exhibited superior length

Model	MLM Loss (↓)	MNLI/-MM	SST-2	STSB	RTE	QNLI	QQP	MRPC	CoLA	GLUE (Avg.) (↑)
AbsPE	2.11	81.7/82.2	91.7	86.6	54.5	88.2	87.1	88.7	46.1	78.5
RelPE	2.10	80.7/81.2	91.9	84.7	61.0	86.9	87.0	88.1	43.0	77.9
NoPE	5.58	62.1/63.2	84.1	72.1	53.8	74.8	82.8	79.6	12.4	64.9
MaskNoPE	2.10	62.9/63.8	84.4	57.6	57.0	73.9	82.2	79.4	1.4	62.3

Table 1: **Performance Comparison of Transformer Variants.** Evaluation of models with different positional encoding strategies: AbsPE (absolute positional encoding), RelPE (relative positional encoding), NoPE (no positional encoding), and MaskNoPE (no positional encoding with causal masks).

generalization in downstream tasks compared to those using widely-adopted absolute and relative positional encoding methods. Given these findings, it becomes compelling to explore strategies for eliminating positional encodings in bidirectional transformers, an avenue that remains largely uncharted to date.

In this paper, we delve into the pretraining of bidirectional LMs using masked language modeling, akin to the BERT-style pretraining, but without the inclusion of positional encoding. Drawing inspiration from the findings of Kazemnejad et al. (2023), we integrate positional information into the model by applying a causal attention mask to the initial layers of the transformer, while retaining full attention in subsequent layers, ensuring the model remains bidirectional. Our experiments with this configuration, which we term **MaskNoPE**, demonstrate that it achieves results comparable to transformers equipped with both absolute and relative positional encodings in masked language modeling tasks. However, its performance falters when assessed on language understanding tasks within the GLUE benchmark, often underperforming significantly. Notably, in some instances, MaskNoPE even yields results inferior to the variant with no positional encoding (**NoPE**).

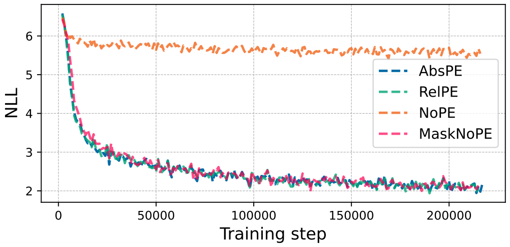

Figure 1: **Masked Languague modeling loss during training.**

2 Experiments & Results

Setup In our exploration of the impact of positional encoding on bidirectional LMs, we assess a range of bidirectional transformer configurations. We consider the **NoPE** variant, which operates without any positional encoding; the **AbsPE** that incorporates absolute positional embeddings; the **RelPE** that utilizes rotary relative positional embeddings; and the **MaskNoPE** variant, which, while eschewing positional encoding, introduces causal masking to the initial layers. All models are pretrained using the masked language modeling (MLM) objective on a subset of the C4 corpus. Our experimental setup follows that of Geiping & Goldstein (2022), employing a 16-layer transformer with an embedding width of 768. All the models were trained for 12 hours on a V100 32G GPU using the AdamW optimizer. We evaluate downstream performance on the GLUE benchmark (Wang et al., 2019) using default hyperparameter settings.

Results As shown in Figure 1, the MLM loss for NoPE, unsurprisingly, plateaus at an elevated value, showing the importance of positional information. Yet, the MaskNoPE setup, leveraging only causal masking, mirrors the performance of both AbsPE and RelPE, all converging at comparable losses. This suggests that causal masks might effectively compensate for the absence of traditional positional information in transformers. However, the narrative shifts on downstream GLUE performance. MaskNoPE trails notably behind the AbsPE and RelPE configurations and is on average even surpassed by NoPE. This disparity underscores the indelible value of positional encodings in bidirectional LMs. Moreover, it hints at a nuanced takeaway: pretraining loss, while informative, isn't a definitive predictor of downstream efficacy.

REFERENCES

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. Convolutional sequence to sequence learning, 2017.

Jonas Geiping and Tom Goldstein. Cramming: Training a language model on a single gpu in one day. *ArXiv*, abs/2212.14034, 2022. URL `https://api.semanticscholar.org/CorpusID:255185900`.

Adi Haviv, Ori Ram, Ofir Press, Peter Izsak, and Omer Levy. Transformer language models without positional encodings still learn positional information, 2022.

Kazuki Irie, Albert Zeyer, Ralf Schlüter, and Hermann Ney. Language modeling with deep transformers. In *Interspeech 2019*. ISCA, sep 2019. doi: 10.21437/interspeech.2019-2225. URL `https://doi.org/10.21437%2Finterspeech.2019-2225`.

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

Ofir Press, Noah A. Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation, 2022.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2020.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding, 2019.

A MODEL

To investigate the impact of positional encodings (PEs) on bidirectional Transformer language models (LMs), we focus on four different configurations: Absolute Positional Encoding (AbsPE), Relative Positional Encoding (RelPE), No Positional Encoding (NoPE), and a modified version with no positional encoding but incorporating causal masks (MaskNoPE). The model's architecture follows the standard Transformer architecture, which first embeds input sequence ids $X = \{x_1, \ldots, x_N\}$ into an embedding dimension D, and then passes them through a series of L Transformer layers:

$$\begin{aligned} \boldsymbol{H}_0 &= \texttt{Word\_Embedding}(X) \\ \boldsymbol{H}_i &= \texttt{Transformer\_layer}_i(\boldsymbol{H}_{i-1}) \quad \text{for} \quad i = 1, \ldots, L \end{aligned} \tag{1}$$

`Word_Embedding` is the word embedding layer and $\texttt{Transformer\_layer}_i$ corresponds to the i-th transformer layer, consisting of a multi-head attention mechanism followed by position-wise feed-forward networks.

NoPE In the NoPE (No Positional Encoding) configuration, we utilize the Transformer architecture as outlined in equation 2, but without incorporating any form of positional encoding. Consequently, this setup is indifferent to the order of the input sequence. Therefore, even if the input sequence is randomly permuted, the output from the Transformer will correspondingly reflect this permutation, i.e maintaining the same values but in the permuted order.

APE For the APE configuration, the initial hidden state $\boldsymbol{H}_0$ is augmented with an absolute positional encoding vector

$$\boldsymbol{H}_0 = \texttt{Word\_Embedding}(X) + \texttt{Pos\_emb}(N) \tag{2}$$

In this configuration, $\texttt{Pos\_emb}(N)$ produced a distinct embedding vector for each position in the sequence, ranging from 1 to N. The positional embedding vectors can be either randomly initialized and subsequently learned during training, or it can utilize a fixed sinusoidal embedding, as proposed in the original Transformer paper.

MaskNoPE In the MaskNoPE configuration, causal masks are applied to the first K layers of the Transformer model, while the remaining layers are left unmasked, i.e., bidirectional. The hidden states are computed as follows:

$$\begin{aligned} \boldsymbol{H}_0 &= \texttt{Word\_Embedding}(X) \\ \boldsymbol{H}_i &= \texttt{Transformer\_layer}_i(\boldsymbol{H}_{i-1}, \boldsymbol{M}) \quad \text{for} \quad i = 1, \ldots, K \\ \boldsymbol{H}_i &= \texttt{Transformer\_layer}_i(\boldsymbol{H}_{i-1}) \quad \text{for} \quad i = K+1, \ldots, L \end{aligned} \tag{3}$$

where $\boldsymbol{M}$ represents the causal mask applied to the first K layers. This mask is designed to prevent each position in the sequence from attending to subsequent positions, thus enforcing a unidirectional, or causal, information flow in these layers. For the subsequent layers $i = K+1, \ldots, L$, the causal mask is not applied, enabling these layers to function bidirectionally. So overall, this model remain bidirectional.