# The Impact of Positional Encoding on Length Generalization in Transformers

Amirhossein Kazemnejad, Inkit Padhi, Karthikeyan Ramamurthy, Payel Das, Siva Reddy

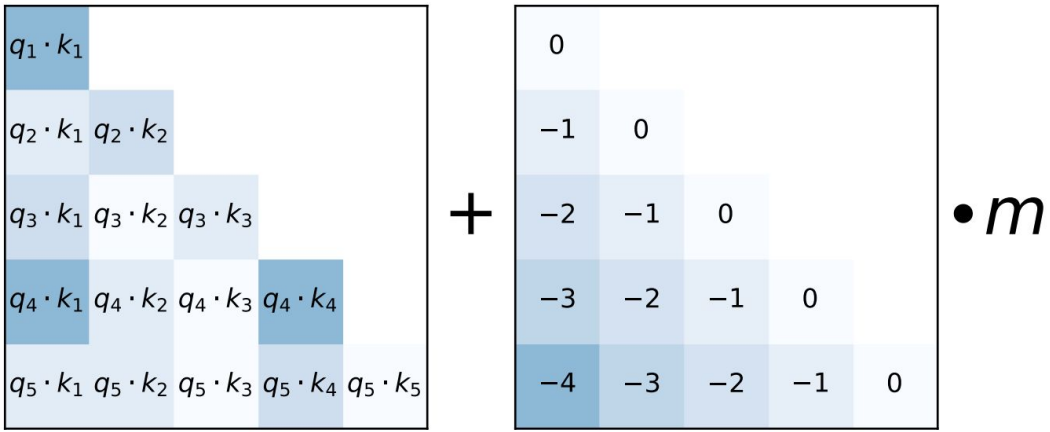
NeurIPS 2023

Presented by: Mohit

**TLDR:** Transformers trained without positional encoding are surprisingly good!

# Paper-specific background

- **“Length generalization”:** The concept that a model trained on sequences of length N can still handle sequences of length > N effectively.
- **“ALiBI” positional encoding (Press et al., 2022):**



# Motivation

**Goal**: Evaluate Transformers trained without any positional encoding and explain how they may still be able to encode relative positions

- **Why does it matter?** If the model can encode positions without any explicit encodings, then it may generalize better to longer sequences

# Approach

**Very simple**: train a bunch of Transformer LMs of the same size, varying only the positional encoding

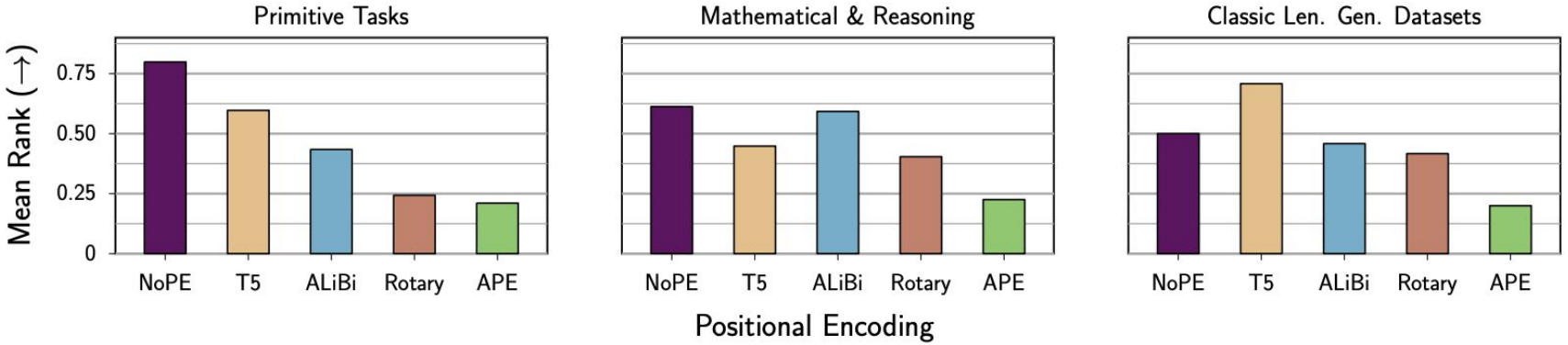
- 107 *million* params (very small!)
- *Synthetic* pretraining data (not natural language!)
  - 100K examples of each task for training

**Positional encodings:** ALiBI, T5, RoPE, absolute, and their method (**NoPE**)

# Synthetic tasks

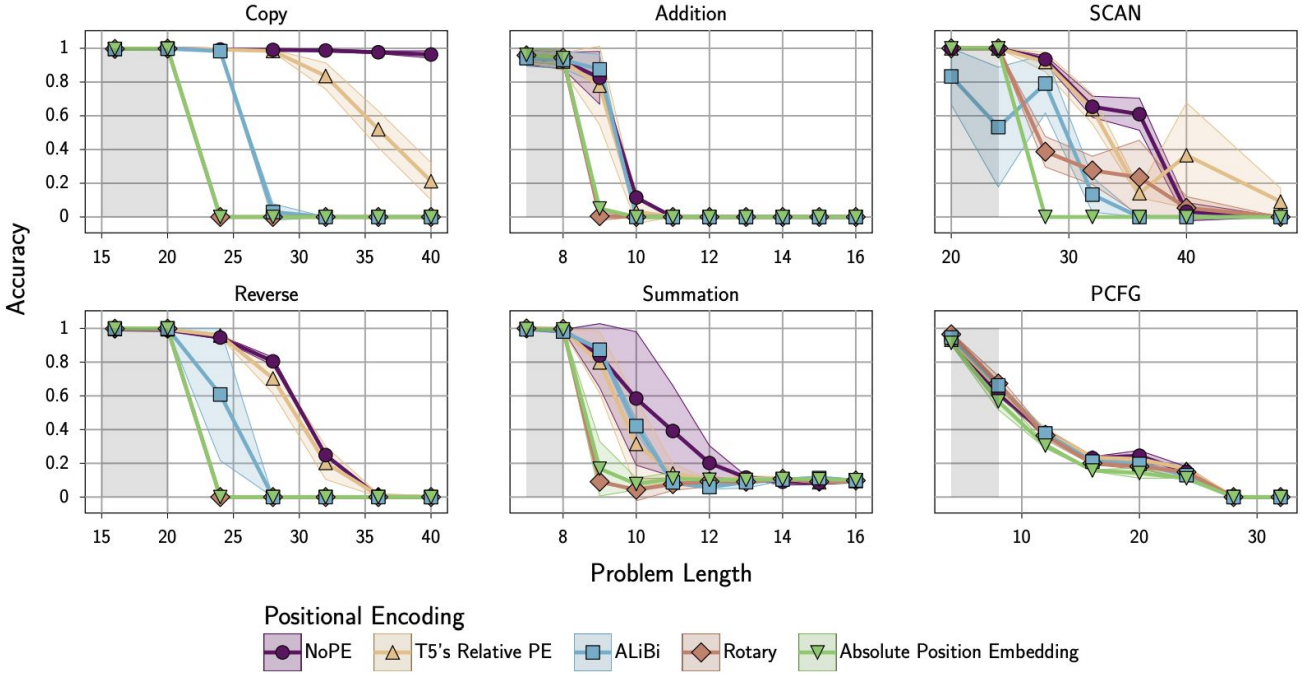
| Task | Input Example | Output Example |
|---|---|---|
| | Primitive Tasks | |
| Copy | `Copy the following words: <w1> <w2> <w3> <w4> <w5>` | `<w1> <w2> <w3> <w4> <w5>` |
| Reverse | `Reverse the following words: <w1> <w2> <w3> <w4> <w5>` | `<w5> <w4> <w3> <w2> <w1>` |
| | Mathematical and Algroithmic Tasks | |
| Addition | `Compute: 5 3 7 2 6 + 1 9 1 7 ?` | `The answer is 5 5 6 4 3.` |
| Polynomial Eval. | `Evaluate x = 3 in ( 3 x ** 0 + 1 x ** 1 + 1 x ** 2 ) % 10 ?` | `The answer is 5.` |
| Sorting | `Sort the following numbers: 3 1 4 1 5 ?` | `The answer is 1 1 3 4 5.` |
| Summation | `Compute: ( 1 + 2 + 3 + 4 + 7 ) % 10 ?` | `The answer is 7.` |
| Parity | `Is the number of 1's even in [ 1 0 0 1 1] ?` | `The answer is No.` |
| LEGO | `If a = -1; b = -a; c = +b; d = +c. Then what is c?` | `The answer is +1.` |
| | Classical Length Generalization Datasets | |
| SCAN | `jump twice and run left` | `JUMP JUMP TURN_LEFT RUN` |
| PCFG | `shift prepend K10 R1 K12 , E12 F16` | `F16 K10 R1 K12 E12` |

# Experiments




**Takeaway**: NoPE is the best-performing position encoding on average!

# Length generalization




**Takeaway**: NoPE exhibits superior length generalization!

# How is this possible?

*TLDR*: Transformers can encode position info even with NoPE!

- You can construct word embeddings and attention weight matrices (e.g., W_q, W_v) such that absolute position is encoded in the hidden state

$$\boldsymbol{W}_E = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ e_{4,1} & e_{4,2} & e_{4,3} & \dots & e_{4,V} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ e_{d,1} & e_{d,2} & e_{d,2} & \dots & e_{d,V} \end{bmatrix}_{d \times V}$$

*Always 1*

*1 if token is <bos>*

*Always 0*

$$\boldsymbol{H}^{(l)} = \begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & 0 & 0 & 0 & \dots & 0 \\ 1 & 2 & 3 & 4 & \dots & T+1 \\ h_{4,1} & h_{4,2} & h_{4,3} & h_{4,4} & \dots & h_{4,T+1} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{d,1} & h_{d,2} & h_{d,3} & h_{d,4} & \dots & h_{d,T+1} \end{bmatrix}_{d \times (T+1)}$$

Armed with absolute encodings, the subsequent layers can encode relative position (shown by similar constructed weight proof)

# Historical predecessor to RoPE

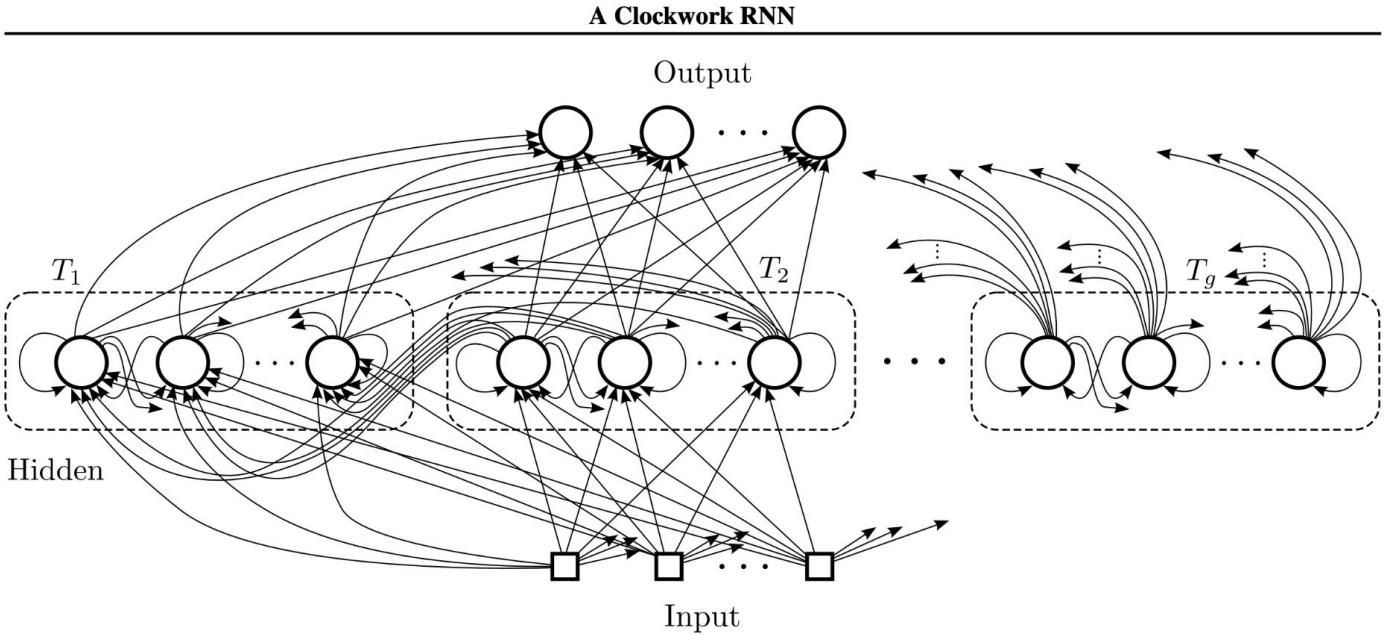
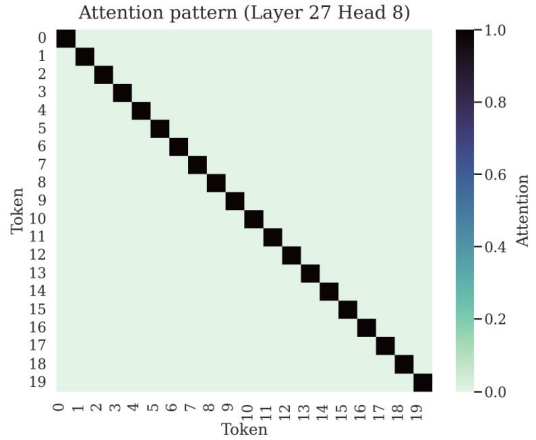



*Figure 1.* CW-RNN architecture is similar to a simple RNN with an input, output and hidden layer. The hidden layer is partitioned into $g$ modules each with its own clock rate. Within each module the neurons are fully interconnected. Neurons in faster module $i$ are connected to neurons in a slower module $j$ only if a clock period $T_i < T_j$.

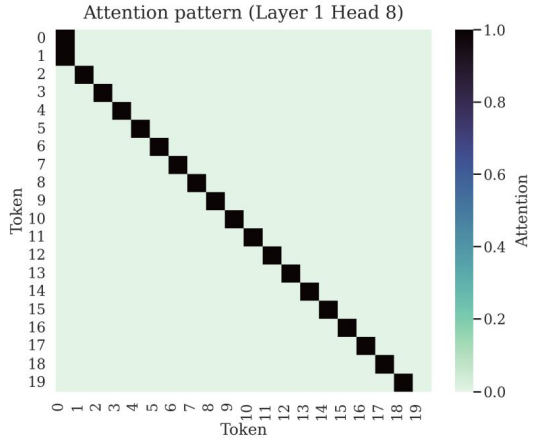
https://proceedings.mlr.press/v32/koutnik14.pdf

# Limitations and open questions

- Did *not* study pretraining on natural language at scale, **does it really work?**
- Theoretical argument depends on universal approximation theorem
  - Impractical? **Can they learn these constructed weights in practice?**
- NoPE theoretically cannot learn purely positional attention heads!




(a) A diagonal head in Gemma 7B.




(b) A previous-token head in Gemma 7B.

See Section 5 of https://arxiv.org/pdf/2410.06205