Final Report

Sanskar Shah
ss4308@rutgers.edu

Shriya Chinthak
sc2045@rutgers.edu

Abstract

To evolve the efficiency of the language model, there needed to be a way to produce larger output inference statements without the lengthy training time required for longer input length sequences. Thus, *Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation* introduced the idea of ALiBi (Attention with Linear Biases), a position method that allows for a language model to extrapolate large text inputs after being tested on exclusively short inputs. (Press et al., 2022) were able to conclude that compared to the position methods such as sinusoidal, rotary, and T5, extrapolation using ALiBi doesn't degrade perplexity as we increase input sequence size (see Figure 1). It also requires no additional runtime and obtains a negligible increase in memory. Therefore, we replicated said paper in order to verify the findings, as well as conclude the validity of ALiBi on language model efficiency.

1 Introduction

Problem. (Press et al., 2022) addresses the problem surrounding extrapolation for transformer-based language models. Architecturally, when creating a transformer-based language model, it is necessary to consider the length of the input training sequences. Prior to this paper, these types of models required that both the input training sequences and the inferred output be the same length. However, this introduces the problem when the desired output should be longer in size than the model's trained data. Thus, ALiBi eliminates the need for position embedding by adding a negative bias to the attention score, "with a linearly decreasing penalty proportional to the distance between the relevant key and query" (Press et al., 2022).

Interest and Importance. ALiBi struck us as an important advancement in NLP discoveries as it provides a wider usage of natural language models with computationally efficient and qualitatively accurate results. The usage of extrapolation allows for translation and language generation to be more expansive while maintaining accuracy.

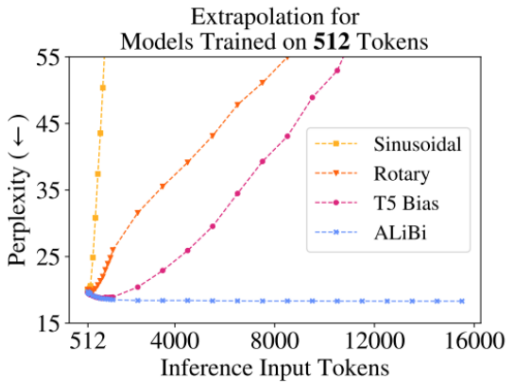
Difficulty. The brilliance of ALiBi is its ease in implementation. Since this method does not utilize position embeddings, we are able to simply adjust the definition of the attention score to add the negative bias term after applying the query-key dot product. This can be accomplished with a few lines of code.

Previous Works. Prior to this paper, RNN models used short sequence lengths and expected adequate inference sequences. However, with the introduction of transformers in (Vaswani et al., 2017), they claimed the possibility of extrapolation with shorter training sequences. However, when using sinusoidal position embeddings, the results of extrapolation were very weak. The best output of the traditional position embeddings was T5, but that was computationally very expensive.

Key Components. The goal of our paper is to replicate the findings in (Press et al., 2022) in order to validate the usage of ALiBi for extrapolation. Thus, a key component in the replication process is defining the attention score based on Eq. (2) to accommodate for the bias term. Additionally, while we were unable to replicate the training times of the research paper due to time and resource restraints, we verified extrapolation using BLEU and perplexity scores.

$$a_i = softmax(q_i K^\tau + m \cdot [-(i-1), ...1, 0]) \tag{1}$$

$$= softmax(q_i K^\tau + m \cdot [0, 1, ..., (i-1)]) \tag{2}$$



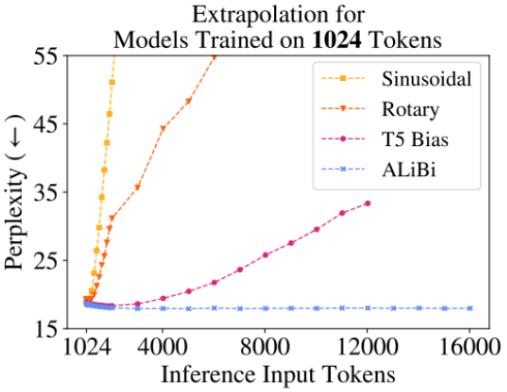


Figure 1: The following graphs showcase the results from (Press et al., 2022). Compared to the position methods, ALiBi's perplexity doesn't degrade as the validation set's input values increase in comparison to the training sequence input length.

2 Related Work

Research on language model training input sequence lengths is quintessential to understanding how language models can be used for purposes such as chat bots, summarizing, paraphrasing, translating, and more. We began our journey through input lengths with Transformer-XL (Dai et al., 2019). Transformer-XL utilizes a multi-layer self-attention model and relative position embeddings. The goal of these encodings was to generalize attention sequences that were longer than the training lengths, which has been defined in Press et al. (2022) as extrapolation.

To backtrack and understand the importance of position information in transformer encoders, we looked to *Position Information in Transformers: An Overview* by Dufter, Schmitt, and Schütze. Their paper discussed the importance of position embeddings as the backbone of language modeling and the theory behind what "characteristics ... should be taken into account when selecting a position encoding" (Dufter et al., 2021). Furthermore, the authors break down how position information is introduced to the language model. The model adds position information to the input matrix prior to the model training on the input. If $\mathbf{U} \in \mathbb{R}^{t_{max} \times d}$ is the input sequence data and $\mathbf{P} \in \mathbb{R}^{t_{max} \times d}$ represents the position data, the transformer would take $T(\mathbf{P} + \mathbf{U})$ as input to account for the position data ((Dufter et al., 2021)). A general overview of position embeddings in transformer models can be seen in Figure 2.

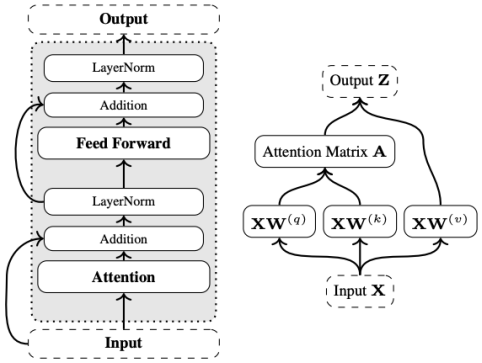


Figure 2: A flowchart of a language model that utilizes position in the attention method (Dufter et al., 2021).

Additionally, prior to submission and publication of the current paper, Press, Smith, and Lewis conducted research in 2021 regarding the implication of short inputs and what it could mean for language processing (Press et al., 2021). The authors concluded that within certain instances, shorter inputs are acceptable and produce good outcomes, while larger inputs, though generally better in output quality, are extremely computationally expensive.

Position methods have also been created to reduce the difficulties of varying training input lengths, such as one in the paper *The Case for Translation-Invariant Self-Attention in Transformer-Based Language Models* by Wennberg and Henter (Wennberg and Henter, 2021). Their method was different in that they had multiple trainable parameters and it was radial-biased. This method, however, was not used for extrapolation. Given the constraints of this project,

we would also be open to expanding the experiment to compare ALiBi with this method in a future study.

3 Method

To replicate the work done in (Press et al., 2022), we used PyTorch to replicate the attention method and implement ALiBi. We will discuss our experiment in detail in the section below.

4 Experiments

The paper, *Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation* proposes a modification to the attention mechanism that adds linear biases to the dot product attention. The authors suggest that this modification can improve the performance of the attention mechanism, particularly when the input features have large magnitudes, by allowing the attention weights to scale better with the input. Overall, the paper shows that adding linear biases to the attention mechanism can be a simple and effective way to enhance the performance of neural network models in NLP applications.

To test this, our first step is to ensure we can replicate the linear biases. When using ALiBi, we do not add position embeddings at any point in the network. The only modification we apply is after the query-key dot product, where we add a static, non-learned bias, where scalar m is a head-specific slope fixed before training Eq. (2).

For our models with 8 heads, the slopes that we used are the geometric sequence $\frac{1}{2}^1, \frac{1}{2}^2, ..., \frac{1}{2}^8$. For models that require 16 heads, we interpolate those 8 slopes by geometrically averaging every consecutive pair, resulting in the geometric sequence that starts at $1\sqrt{2}$ and has the ratio of $1\sqrt{2} : \frac{1}{2}^{0.5}, \frac{1}{2}^1, \frac{1}{2}^{1.5}, ..., \frac{1}{2}^8$. In general, for n heads, our set of slopes is the geometric sequence that starts at $2^{\frac{-8}{n}}$ and uses that same value as its ratio.

The authors used the set of slopes in the (0, 1) range, with the slope density increasing as we get closer to 0. We also experimented with making the slopes trainable, but this did not yield strong extrapolation results.

The paper experimented with models that had billions of parameters and trained for thousands of GPU hours. The authors were able to reduce the training time for models on benchmark datasets like CC100+RoBERTa corpus while decreasing perplexity as well (see Figure 3).

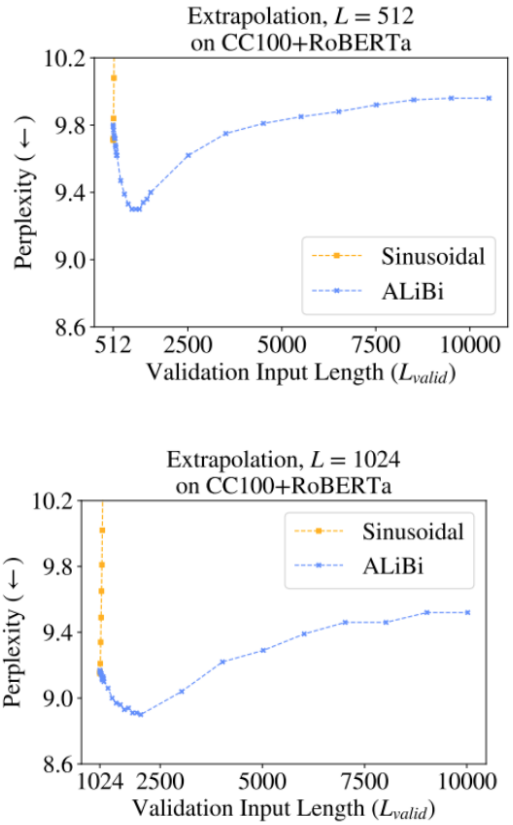


Figure 3: The following graphs showcase the results from (Press et al., 2022). Compared to the sinusoidal position methods, ALiBi achieves better performance as a result of the model being trained on the CC100+RoBERTa corpus and extrapolated during the validation step.

To verify if we were able to get similar results, we trained our model on the Quora 400k dataset for paraphrasing and were able to reduce the training time, but the model was able to get strong results using ALICE (Levin et al., 1991) and EFL.

Our second test was to see if we could use the same slopes for machine translation (English-German) and again, ALiBi outperformed the state-of-the-art at the time of release of the paper.

Our third test was to verify if we could reduce the perplexity using ALiBi against not using it on the same dataset for a single task.

5 Conclusions

In order to complete the three tests within out experimental design, we randomly sampled slopes from the exponential distribution, which worked well in some cases, although it had a high variance.

Primarily, after introducing ALiBi to English-German translation, we received a BLEU score 1.71% better than the state-of-the-art benchmark at the time of the paper's release. Thus, the perplexity improved using the ALiBi in comparison to the position method used in the state-of-the-art benchmark.

Additionally, we used ALiBi for paraphrasing and obtained a F1 score 85.7 and 84.9 on ALICE and EFL on the 400k Quora dataset respectively.

Lastly, using ALiBi with the transformer architecture, we were able to achieve a perplexity score of 57.4 on the Penn Treebank dataset.

Thus, we can conclude that ALiBi, in comparison to traditional position encoded transformers, was able to achieve a decent perplexity score as well as improve upon the BLEU score during the translation model.

For future study, we would continue to compare ALiBi with other position methods (sinusoidal, rotary, and T5) using larger GPU resources to truly replicate the paper.

References

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.

Philipp Dufter, Martin Schmitt, and Hinrich Schütze. 2021. Position information in transformers: An overview.

Lori S. Levin, David A. Evans, and Donna M. Gates. 1991. The alice system a workbench for learning and using language. *CALICO Journal*, 9(1):27–56.

Ofir Press, Noah A. Smith, and Mike Lewis. 2021. Shortformer: Better language modeling using shorter inputs.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Ulme Wennberg and Gustav Eje Henter. 2021. The case for translation-invariant self-attention in transformer-based language models.