University of California San Diego

Course #: CSE 256

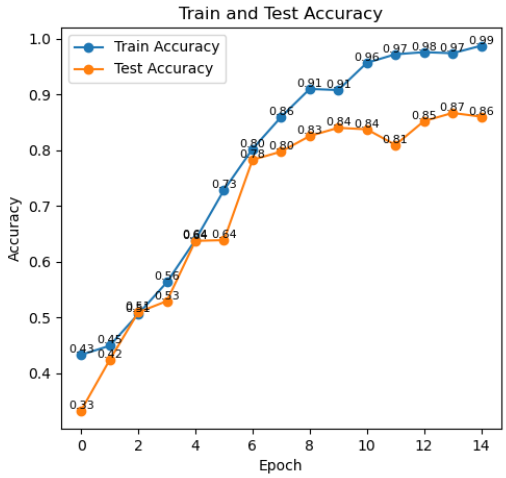
PA2: Transformer

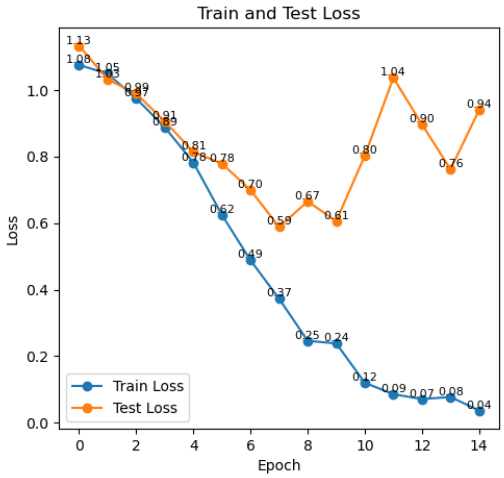
November 7, 2024

Author	*Student ID*
Jingyu Wu	A16157847

1 Part 1: Encoder Trained With Classifier

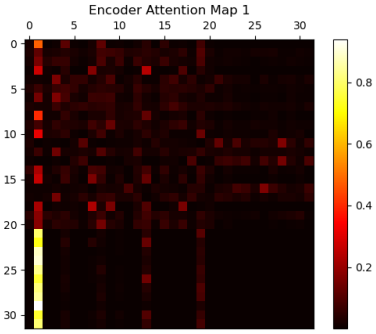
In this part, I have implemented the Transformer Encoder along with a Feedforward Classifier and trained them simultaneously for the encoder to learn representations that are specifically useful for the speech segment classification task. The number of parameters in the model is 576467. With the default hyperparameters, here are the results:

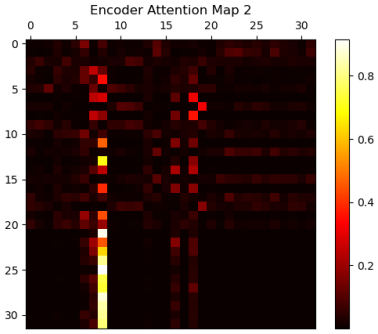


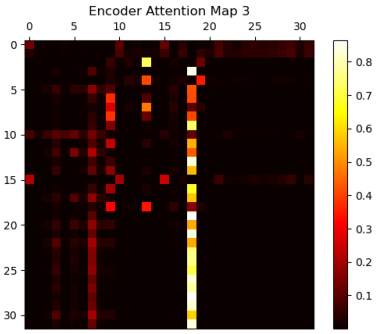


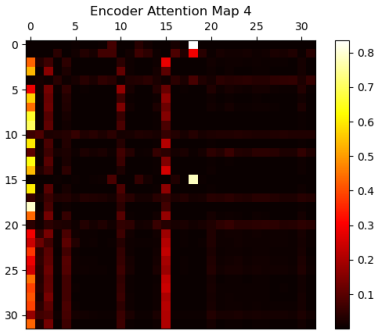
As shown in the plots, the model converges on the training set, reaching almost 100% accuracy, but generalizes less well on the testing set, achieving only 86.4% final accuracy. The losses show similar trends. After 6 epochs, the test loss stops decreasing, indicating possibilities of over-fitting.

I also took out the attention matrix of the first head from each attention layer:





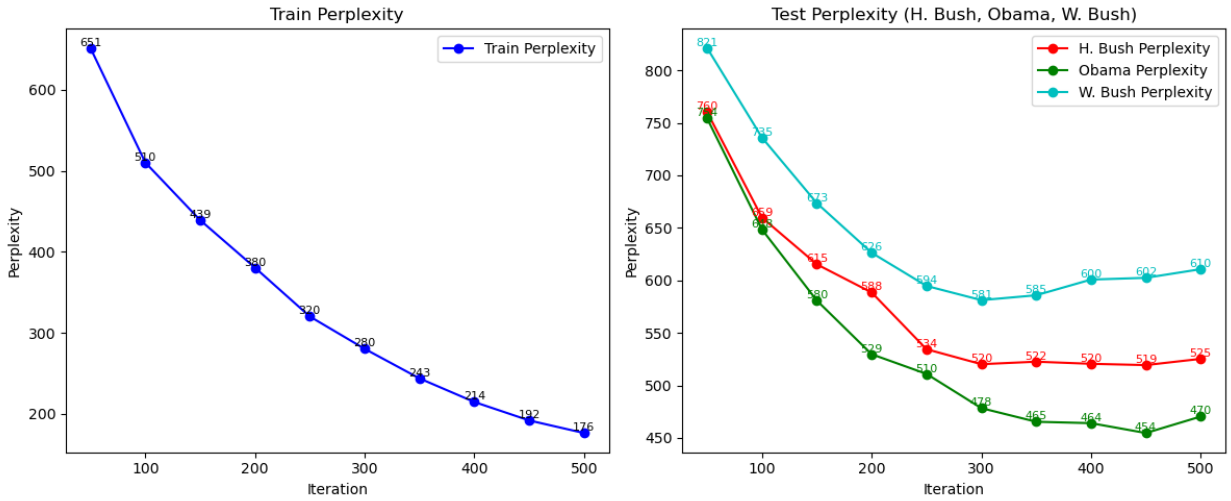




The example I chose was "In fact, I will be right there with you, as a citizen, for all my remaining days." As shown in those heat maps, each layer of the model seems to focus on different part of the input. The first layer emphasizes on the second token, which is the embedding of "fact" in the example sentence, indicating possibilities of more usage of the word "fact" for the speaker. The higher layers might be focusing more about abstract semantic information.

2 Part 2: Pretraining Decoder Language Model

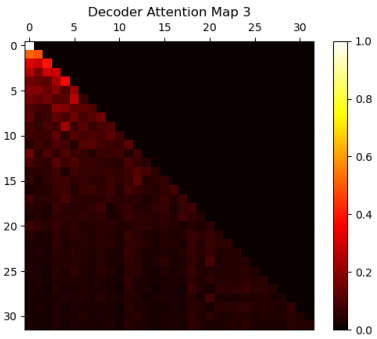
In this part, I have implemented the Transformer Decoder with masked attention along with a Feedforward Classifier. Then I pretrained them on the training text by trying to predict the next word in a sequence given the previous words. The number of parameters in the model is 1157291. With the default hyperparameters, here are the results:

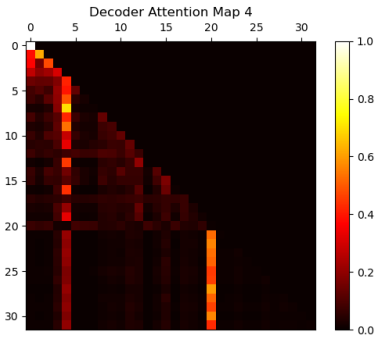


As shown in the plots, the perplexity keeps dropping on the training set, but for the test sets, the perplexity stops decreasing after about 300 iterations. The final perplexity for the training is 176, and the perplexity for the test sets are 525, 470, and 610 respectively for H. Bush, Obama, and W. Bush.

I also took out the attention matrix of the first head from each attention layer:

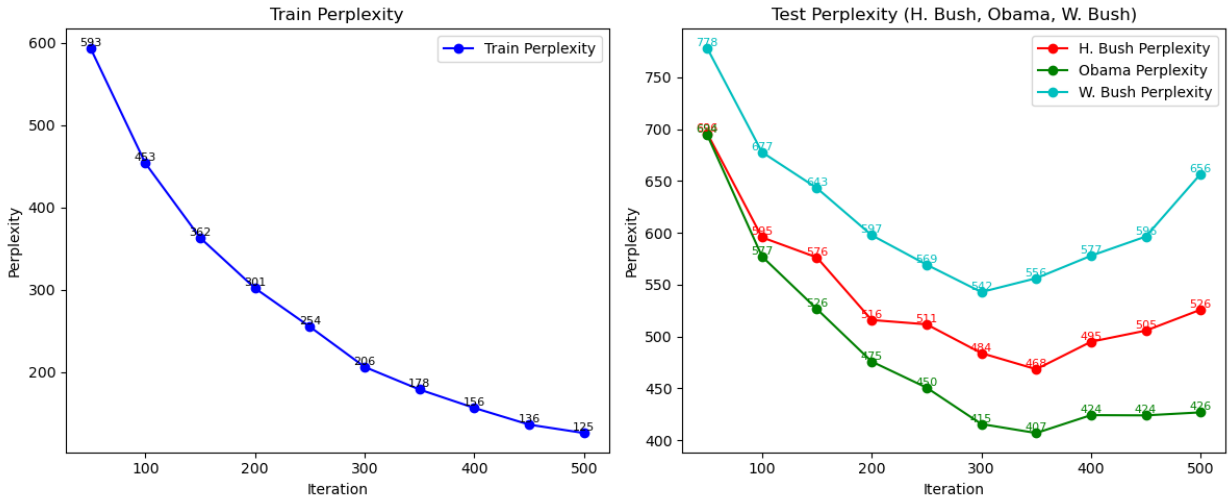
The attention maps for the first few layers seem rather sparse, probably because the transformer is trying to extract information from tokens among the whole sentence seen so far. The attention map for the last layer is rather focused on two particular positions, which likely contain important information for the transformer to make decisions about the next token to predict.





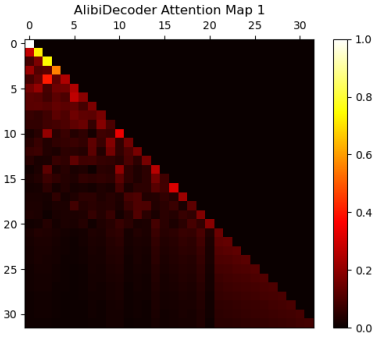
3 Part 3: AliBi Positional Encoding

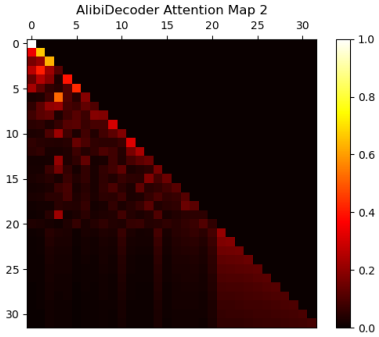
To explore alternatives to the positional embedding, I also implemented AliBi, a positional bias method for transformers that applies linearly increasing attention biases based on token distance, enabling the model to focus on relative positions without explicit positional embeddings. The number of parameters in the model is 1155243. With same hyperparameters, here are the results on the decoder with AliBi:

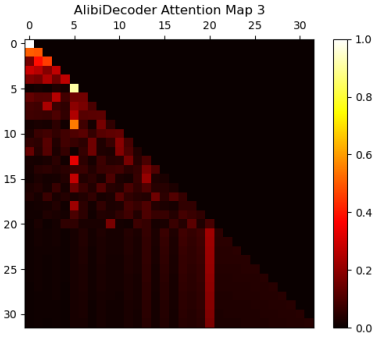


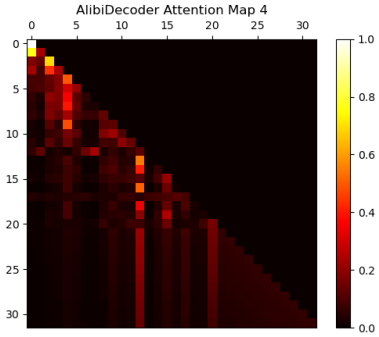
Compared with regular positional embedding, AliBi achieves a much lower training perplexity (125 vs 176), similar testing perplexity on the H. Bush dataset (526 vs 525), a lower perplexity on Obama dataset (426 vs 470) and a much higher perplexity on W. Bush dataset(656 vs 610). The increased perplexity on W. Bush dataset may be caused by overfitting into the training dataset as shown by all three testing curves.

I also took out the attention matrix of the first head from each attention layer:









As expected, we see higher values near the main diagonal, meaning that we tend to pay more attention to tokens closer to the current position than those further away.

References

[1] Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. CoRR, abs/2004.05150, 2020.

[2] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Polosukhin, I. (2017). Attention is All You Need. arXiv preprint arXiv:1706.03762.

[3] Andrej Karpathy. (2023, Jan 17).Let's build GPT [Video]. YouTube. https://www.youtube.com/watch?v=kCc8FmEb1nY