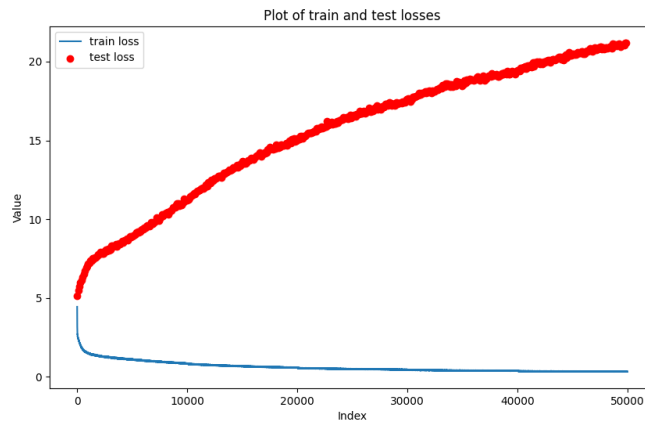


Assignment 3: Transformer Language Models

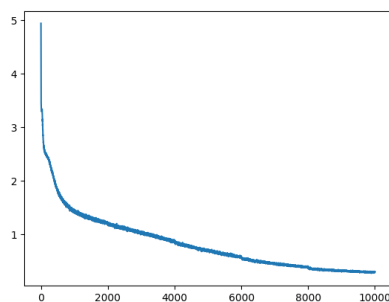
1. We quickly noticed that we could achieve arbitrarily low loss after enough training batches. So, a big challenge was achieving low loss while not over fitting. **First**, via continuous sampling, we noticed that networks with low enough loss are returning sentences from the training data, not generating new ones. (Side note: we changed the prompt from “hello” to “my lord” for the Shakespeare segment). **secondly**, we tried splitting the data to test and train parts, this enabled us to visually see the overfit. The graph below shows the loss on the train data vs the loss on the tests data, expressing the perplexity of the model on the test data



We concluded that in the scope of this work we did not have enough tools to determine a stopping point for all train scenarios. In the baseline experiments, the sampling (done with *better sampling* suggested practices) was stable and made sense after training on 7500 batches of 128 samples in each batch with sequence length of 128, summing to seeing 122,880,000 characters. Since arbitrarily low loss could be achieved, we chose 122,880,000 characters point to be our stopping point for training.

Our best results of loss 0.42 (with the shakespeare data) were thus achieved after 7500 batches, with a network with 5.42 million parameters. It comprised of with batch size of 128, 12 layers, 6 attention heads, 192 embedding dimension size (and $768=4 \times \text{hidden dim size of hidden MLP layer size}$), scheduler for the learning rate and weight decay usage of 0.01.

See loss curve for shakespeare data (we got the model from the 7500 mark):



Roe esquira, ID 309840791

Yedidia Kfir, ID 209365188

Again we remind that we could get arbitrarily low loss after enough epochs, but overfit and quality of sampling guided us to choose this model.

2.

- a. Experiments with **heads** number of 4,6,8,12,16. 6 heads showed the best loss at the stopping point. Also, with 8, 12, 16 heads we saw bad results in the sampling, producing words with wrong spelling and also higher loss. Thus, we chose 6 to be the head number.
- b. Experiments with layers number of 6, 12, 24. 12 layers produced the best loss, yet all produced high quality sampling. Thus we chose 12 to be the layer number.
- c. Experiments with embedding dimensions of 192, 252, 480. 480 embedding showed the lowest loss at the stopping point with 0.3 loss, but the sampling results showed spelling mistake and non-comprehensible text. Dimension size 252 also showed bad sampling results, although showing low loss. We concluded using 192 as embedding dimension size.

3. Improvements:

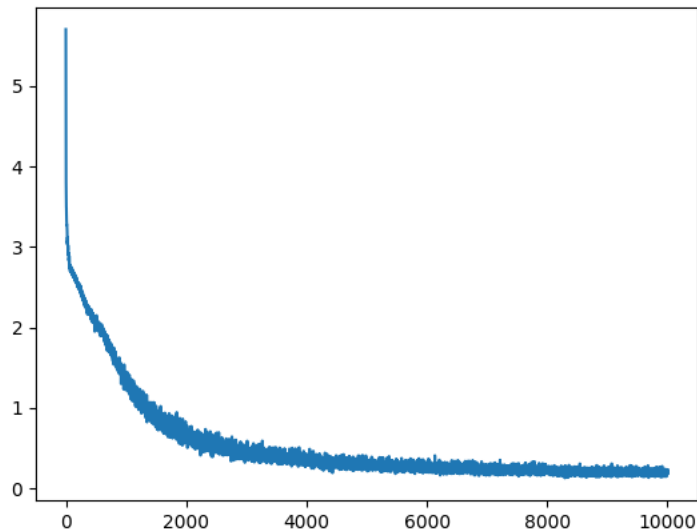
- a. We added Xavier Glorot normal initialization for all matrix parameters. This helped convergence.
- b. We added weight decay of rate 0.01.
- c. We added scheduler for learning rate auto adjustment.
- d. We increased batch size to 128, this improved loss. Also, we tested higher batch size but the affect was not significant.
- e. We tried adding dropout layers as suggested but we found out that using weight decay and scheduler yielded better results and adding dropout layers on top of these components damaged performances.

4. The hebrew data enabled us to better assess the samples since for us it is more accessible then Shakespearian english. The prompt we used was "חיים נחמן ביאליק" in order to get a full poem or two. We used the same parameters as the Shakespeare data with 7500 batches, with a network with 5.42 million parameters. It comprised of with batch size of 128, 12 layers, 6 attention heads, 192 embedding dimension size (and $768=4*\text{hidden dim size of hidden MLP layer size}$), scheduler for the learning rate and weight decay usage of 0.01. **this yielded a loss of 0.22**, which is about half the loss with the Shakespeare data. This could be attributed to lower variance in the Hebrew data, maybe due to the fact that the Hebrew data are poems, and the Shakespeare data is segments of a one or more plays. Also, the language models were trained on a character base, and Hebrew is more compact language per character. In English there are a lot of vowels usage while in Hebrew it can be avoided, aspecially in high level Hebrew found in the poems of Rachel and Bialik.

Roe esquira, ID 309840791

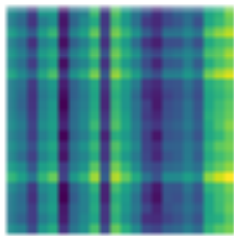
Yedidia Kfir, ID 209365188

See loss curve for hebrew data (we got the model from the 7500 mark):



5. To analyze the weights of the model. We created a heat map on the self-attention of each layer and each head. The map indicates how each word relates to other words in the sentences.

First, we observed an interesting pattern: the model demonstrates a distinct capability to discern and individually highlight words, assigning unique attention weights to each. This selective attention allows the model to focus on specific words that are crucial for understanding the sentence's structure and meaning.



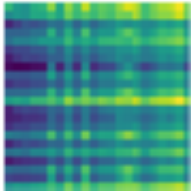
In the above example, for example, the attention is focused around the beginning of the sentences, specifically on the word “Thou” which is the subject of the sentence. By concentrating on “Thou,” the model is likely trying to capture the essence of the sentence.

Additionally, we observed that the map is very strong at the end of the sentence, especially in the early layers, as exemplified below:

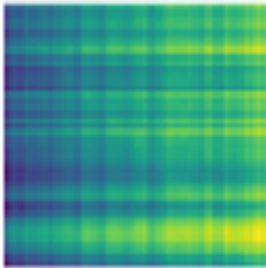
Roe esqira, ID 309840791

Yedidia Kfir, ID 209365188

Layer 1 Head 1



This may indicate the structure of the sentences in the data (the Shakespeare data) which puts heavy weight on the end of the sentences to induce drama in the audience. We see the reverse in the Hebrew model, where the subject often appears at the start of the sentence, so does the attention, as we can see in the following image:



Note that Hebrew is read from right to left, so here the attention is at the start of the sentence (juxtaposed to the previous images where the attention is more toward the end).

6. The project was very nice and easy to follow. The major obstacle was overcoming overfit problem in a domain we are not familiar with (high literature). Further elaboration on this matter is found in section 1.