Joseph Blom and Grayson Snyder

MSAI 337: Deep Learning for Natural Language Processing

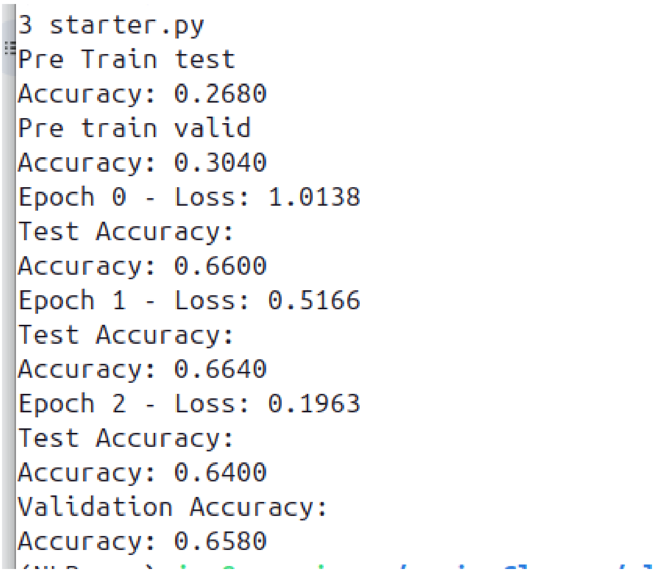
Professor David Demeter

11 May 2025

Homework 2 Report

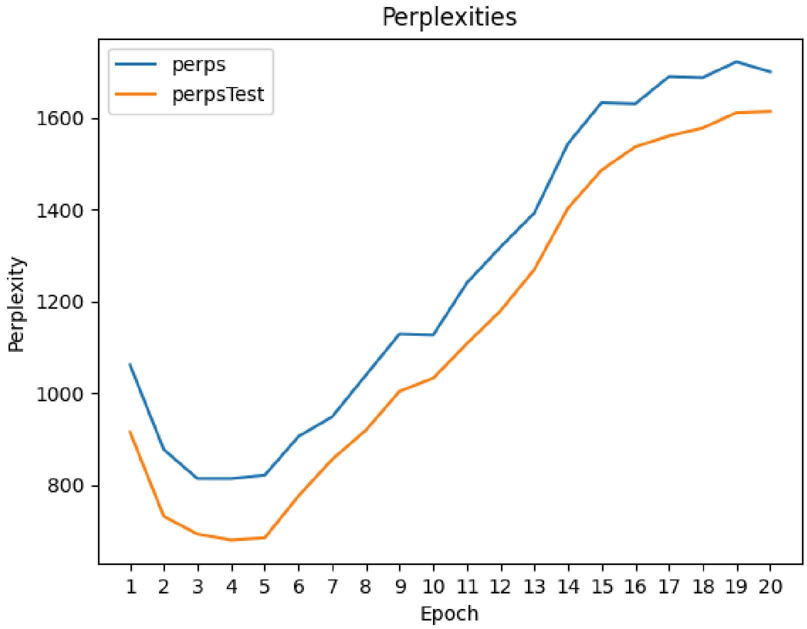
1) For part one we generally tried to follow the setup laid out in the assignment. We used the encoding setup with the CLS and END tokens as described. We ended up using a refactored version of the file parsing code provided, however the data shape is the same under the hood. We then take the dot product with the top context embedding and a linear layer and apply a softmax as part of a super standard nn.CrossEntropyLoss(). As we can see below, the bert uncased model out of the box works pretty poorly, however the performance improves dramatically after a short, 3 epoch training loop. We used ChatGPT to diagnose some issues with packaging our input data as a data loader as well as to clean up our code.

   Like the last assignment, some of our biggest issues came in the shape of loading the data into the model. Turning the input data from our file into a form that our model could use was challenging and while we would have liked to avoid using the data loader type as it confuses things we were able to get things working well with some AI assistance.

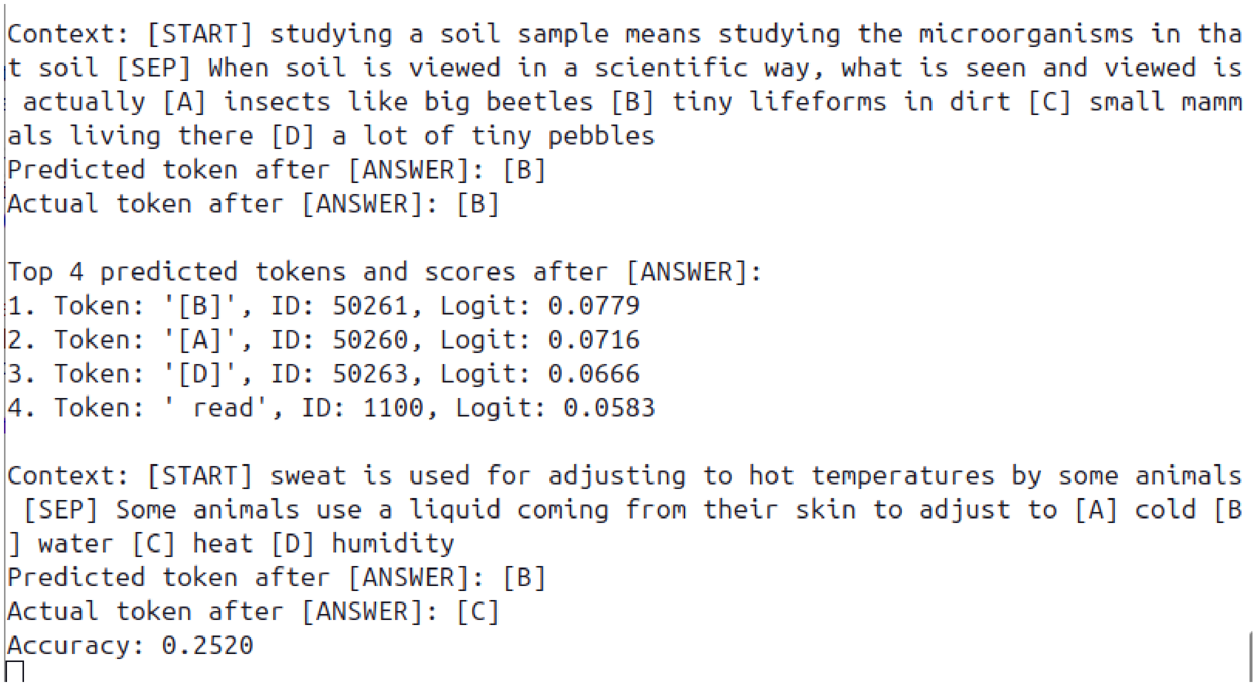


```
3 starter.py
Pre Train test
Accuracy: 0.2680
Pre train valid
Accuracy: 0.3040
Epoch 0 - Loss: 1.0138
Test Accuracy:
Accuracy: 0.6600
Epoch 1 - Loss: 0.5166
Test Accuracy:
Accuracy: 0.6640
Epoch 2 - Loss: 0.1963
Test Accuracy:
Accuracy: 0.6400
Validation Accuracy:
Accuracy: 0.6580
```

2) To approach the multiple choice questions with a generative approach, we began with our code from Homework 1. For this problem, the GPT2TokenizerFast is used as the tokenizer with special tokens added for the format of the data making used of [START], [SEP], [ANSWER], [A], [B], [C], and [D] tokens. Next, generate Datasets to be wrapped in DataLoaders for the three jsonl files provided. To get the model, a Transformer is created with the vocabulary size and hyperparameters, then the pretrained weights from the Wikitest-103 model are loaded. There is discrepancy between the sizes of these due to the addition of the tokens in the encoding of the data, so the pretrained weights and biases must have the additional 7 incorporated using the mean of the already existing embeddings to stitch them together at an appropriate size. The model training is essentially the same as the previous homework, just with new information. For each epoch, the batches and a tuple of input ids and attention mask are enumerated to loop over. The outputs are generated with a forward pass through the model, then the loss is computed for the token after [ANSWER] which should be A, B, C, or D. There is then a backward pass of the loss and step of the optimizer before continuing for the next batch.



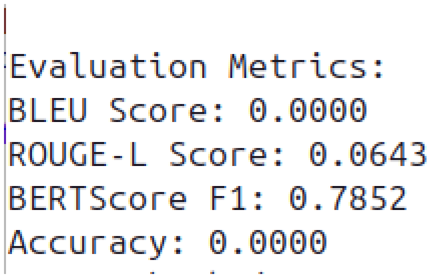
The final perplexity for the Wiki103 pre-training was 1700.



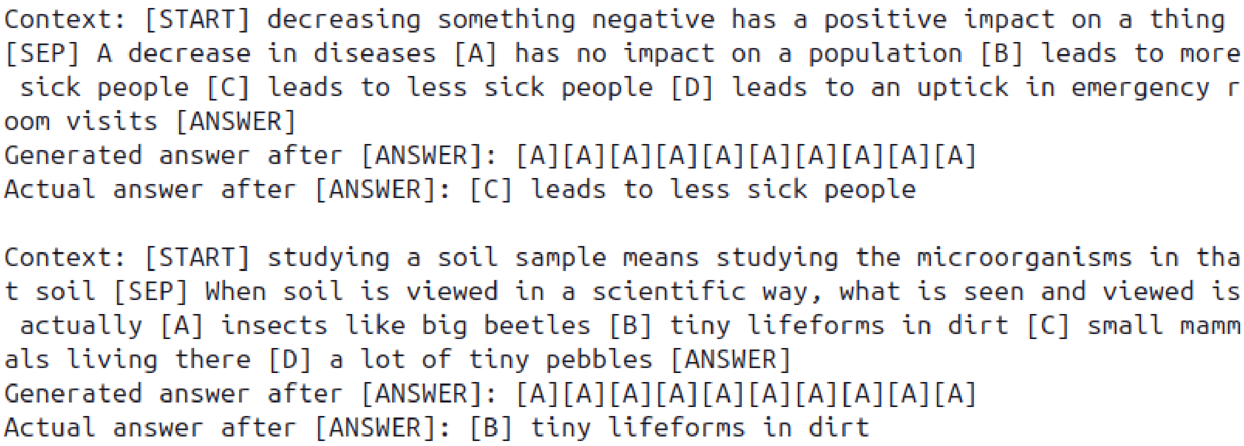
```
Context: [START] studying a soil sample means studying the microorganisms in tha
t soil [SEP] When soil is viewed in a scientific way, what is seen and viewed is
 actually [A] insects like big beetles [B] tiny lifeforms in dirt [C] small mamm
als living there [D] a lot of tiny pebbles
Predicted token after [ANSWER]: [B]
Actual token after [ANSWER]: [B]

Top 4 predicted tokens and scores after [ANSWER]:
1. Token: '[B]', ID: 50261, Logit: 0.0779
2. Token: '[A]', ID: 50260, Logit: 0.0716
3. Token: '[D]', ID: 50263, Logit: 0.0666
4. Token: ' read', ID: 1100, Logit: 0.0583

Context: [START] sweat is used for adjusting to hot temperatures by some animals
 [SEP] Some animals use a liquid coming from their skin to adjust to [A] cold [B
] water [C] heat [D] humidity
Predicted token after [ANSWER]: [B]
Actual token after [ANSWER]: [C]
Accuracy: 0.2520
```

The generative approach worked much worse compared to the classification approach from part 1. The generative approach of producing only the letter results in only one letter being generated recurrently. As the model continued training, the weights of each letter option became very close, and whichever happened to be slightly above the others would end up being the only selected answer. This was the key limitation in the model's success. It is possible that using a better pretrained model would alleviate this problem, as we did not have the chance yet to use the provided HW1 model and were using our own shoddy model. Professor Demeter advised that we go ahead and submit this now for grading based on our analysis, and we are using the provided model to attempt to produce better results for this and future work.

3) The setup for this problem is mostly the same as in problem two. The main difference comes in the generation of the output, as part 2 only required the correct letter while this problem seeks an output of the entire correct answer. The zero-shot and fine-tuned accuracies are both zero.



```
Evaluation Metrics:
BLEU Score: 0.0000
ROUGE-L Score: 0.0643
BERTScore F1: 0.7852
Accuracy: 0.0000
```



```
Context: [START] decreasing something negative has a positive impact on a thing
[SEP] A decrease in diseases [A] has no impact on a population [B] leads to more
 sick people [C] leads to less sick people [D] leads to an uptick in emergency r
oom visits [ANSWER]
Generated answer after [ANSWER]: [A][A][A][A][A][A][A][A][A][A]
Actual answer after [ANSWER]: [C] leads to less sick people

Context: [START] studying a soil sample means studying the microorganisms in tha
t soil [SEP] When soil is viewed in a scientific way, what is seen and viewed is
 actually [A] insects like big beetles [B] tiny lifeforms in dirt [C] small mamm
als living there [D] a lot of tiny pebbles [ANSWER]
Generated answer after [ANSWER]: [A][A][A][A][A][A][A][A][A][A]
Actual answer after [ANSWER]: [B] tiny lifeforms in dirt
```

The second version of the generative model had similar issues to the first. As with the generative model that produced the answer choice, this new generative model does a similar thing, but attempts to generate an answer. What ultimately happens is the answer generated is just the character it chose over and over, for example “[B][B][B][B]...” So again, it is possible that it is due to the poorly pre-trained model we are using. These results are completely useless compared to any previous results.