NLP Assignment 3

Name: Vamsi Tallam UIN: 432001932

1. Text Generation Basics:

Task1: How to run the code:

1. Download and extract the code from PA3-text-generation-1.zip
2. Upload the folder on google drive and change the current working folder to the PA3-text-generation.
3. Install dependencies (!pip install transformers and !pip install datasets)
4. Prompt.txt can have the following sentences (one at a time):
    a. Text generation is the task of generating text with the goal of
    b. Be the change that you want to see
    c. These violent delights have violent ends. And
    d. I try to know as many people
5. Python generation.py -m 25 to run the code

Results and Analysis:

The following are the results from the basic text generation using the prompts mentioned above.

```
[15] !python generation.py -m 23

1
Task 1: Sentence generation
==============
Text generation is the task of generating text with the goal of generating a text file.

The following example shows how to generate a text file with the following command:
```

```
[16] !python generation.py -m 25

1
Task 1: Sentence generation
==============
Be the change that you want to see in your life.

I'm not saying that you should be a "good" person. I'm saying that you
```

```
[18] !python generation.py -m 21

1
Task 1: Sentence generation
==============
These violent delights have violent ends. And they are not just for the sake of the people. They are for the sake of the world.
```

```
[20] !python generation.py -m 23

1
Task 1: Sentence generation
==============
I try to know as many people as possible about the situation, and I try to keep my eyes open for any information that might be helpful.
```

- **Prompt:** “Text generation is the task of generating text with the goal of”
  Result: “generating a text file. The following example shows how to generate a text file with the following command:”
    - A simple google search of the prompt results with the following:
        - Text generation is the task of generating text with the goal of appearing indistinguishable to human-written text. This task if more formally known as "natural language generation" in the literature.
    - The generated text(result) as we see is clearly different from the actual text.
    - The generated text is cohesive and has a clear connection between the two sentences. The grammar of the text is good for the most part.

- **Prompt:** “Be the change that you want to see”
  Result: “in your life. I'm not saying that you should be a "good" person. I'm saying that you”
    - A simple google search of the prompt results with the following:
        - Mahatma Gandhi famously said, "You must be the change you want to see in the world."
    - The generated text(result) as we see is a little different from the actual text. However, we can see that the resulting context is similar. The text is cohesive and grammatically correct.

- **Prompt:** “These violent delights have violent ends. And”
  Result: “they are not just for the sake of the people. They are for the sake of the world.”
    - A simple google search of the prompt results with the following:
        - “These violent delights have violent ends. And in their triumph, like fire and powder Which, as they kiss, consume” — William Shakespeare, Romeo and Juliet
    - The generated text(result) as we see is different from the actual text. However, we can see that the resulting context is not similar. The text is not completely cohesive and grammatically correct.

- **Prompt:** “I try to know as many people”
  Result: “as possible about the situation, and I try to keep my eyes open for any information that might be helpful”
    - A simple google search of the prompt results with the following:
        - “I try to know as many people as I can. You never know which one you’ll need."
    - The generated text(result) as we see is different from the actual text. However, we can see that the resulting context is similar. The text is not completely cohesive and grammatically incorrect.

Known Issues and Limitations:

- As we generate a lot of text, we might end up with text which has a completely different context (drift).
- Text generated is not always coherent and grammatically correct.

Task2: Better generations:

- I tried several techniques including greedy decoding, multinomial sampling, Beam-search sampling, Diverse beam-search sampling, and others. Experimented with various lengths and tweaked several hyperparameters like temperature.
- As an example lets observe the results for the following prompt:

Results and Analysis:

Prompt: “Be the change that you want to see”

```
!python generation.py -m 20

1
Task 1: Sentence generation
==============

 Greedy Sampling:
Be the change that you want to see in your life.

I'm not saying that you should be a "good" person.

 Multinomial Sampling:
Be the change that you want to see happen.

"The biggest challenge with this move is I want to get rid of what they

 Beam-search Multinomial Sampling:
Be the change that you want to see in the future?

I think the most important thing is to make sure that you have a

 Beam-search Sampling:
Be the change that you want to see in your life.

If you want to be a better person, you need to change your

 Diverse Beam-search Sampling:
/usr/local/lib/python3.7/dist-packages/transformers/generation_beam_search.py:199: UserWarning: Passing `max_length` to BeamSearchScorer is deprecated
  "Passing `max_length` to BeamSearchScorer is deprecated and has no effect. "
Be the change that you want to see in your life.

I'm not going to lie, I'm not going to lie.

 Diverse Beam-search Sampling with temperature:
Be the change that you want to see in your life.

I'm not going to lie, I'm not going to lie.
```

- The results indicate that the results produced are much better than that of task 1.
- Beam-search sampling for the above prompt produces the closest text in terms of context and also the produced text is coherent and grammatically correct.

Prompt: “I try to know as many people”

```
!python generation.py -m 20

Downloading: 100% 1.04M/1.04M [00:00<00:00, 2.41MB/s]
Downloading: 100% 456k/456k [00:00<00:00, 1.31MB/s]
Downloading: 100% 665/665 [00:00<00:00, 439kB/s]
Downloading: 100% 548M/548M [00:13<00:00, 39.3MB/s]
1
Task 1: Sentence generation
==============

 Greedy Sampling:
I try to know as many people as possible about the situation, and I try to keep my eyes open for any information that might be

 Multinomial Sampling:
I try to know as many people as I can on my phone. And of course, I just try to have a way of finding

 Beam-search Multinomial Sampling:
I try to know as many people as possible about the things that I am doing, so I don't have to tell them every day

 Beam-search Sampling:
I try to know as many people as I can about what's going on in the world, but I don't know what's going

 Diverse Beam-search Sampling:
/usr/local/lib/python3.7/dist-packages/transformers/generation_beam_search.py:199: UserWarning: Passing `max_length` to BeamSea
  "Passing `max_length` to BeamSearchScorer is deprecated and has no effect. "
I try to know as many people as I can, and I try to be as honest as I can, and I try to be

 Diverse Beam-search Sampling with temperature:
I try to know as many people as I can, and I try to be as honest as I can, and I try to be
```

- The results indicate that the results produced are much better than that of task 1.
- Beam-search multinomial sampling for the above prompt produces the closest text in terms of context and also the produced text is coherent and grammatically correct.

Prompt: These violent delights have violent ends. And

```
!python generation.py -m 20

1
Task 1: Sentence generation
==============

 Greedy Sampling:
These violent delights have violent ends. And they are not just for the sake of the people. They are for the sake of the world.

 Multinomial Sampling:
These violent delights have violent ends. And so, on the day of the Feast of Bannister on March 24 in the city of P

 Beam-search Multinomial Sampling:
These violent delights have violent ends. And I don't mean that in a negative way. I mean that in a positive way.



 Beam-search Sampling:
These violent delights have violent ends. And if you're going to do something about it, you're going to have to do something about it

 Diverse Beam-search Sampling:
/usr/local/lib/python3.7/dist-packages/transformers/generation_beam_search.py:199: UserWarning: Passing `max_length` to BeamSearchScorer is deprecated and has no effect.
  "Passing `max_length` to BeamSearchScorer is deprecated and has no effect. "
These violent delights have violent ends. And they are not the end of the world. They are the beginning of the end of the world.

 Diverse Beam-search Sampling with temperature:
These violent delights have violent ends. And they are not the end of the world. They are the beginning of the end of the world.
```

- The results indicate that the results produced are much better than that of task 1.
- Diverse Beam-search sampling with temperature for the above prompt produces the closest text in terms of context and also the produced text is coherent and grammatically correct.

Known Issues and Limitations:

- Hyperparameter tuning produced great results, the text generated is more coherent and has better context.

Task3: Evaluations:

1. Examples and the definition from the link provided are closely observed.
2. To run the code python generation.py –ppl
3. The code was run on gpu and the stride of 128, 256, and 512 were experimented.

Results and Analysis:

Stride: 512

```
!python generation.py --ppl

1
Task 2: Perplexity score
==============
WARNING:datasets.builder:Found cached dataset wikitext (/root/.cache/huggingface/datasets/wikitext/wikitext-2-raw-v1/1.0.0/a241db52902eaf2c6aa732210bead40c090019a499ceb13bcbfa3f8ab646a126)
Token indices sequence length is longer than the specified maximum sequence length for this model (287644 > 1024). Running this sequence through the model will result in indexing errors
100% 560/562 [01:05<00:00,  8.52it/s]
Perplexity = 25.1704.
```

Stride: 256

```
!python generation.py --ppl

1
Task 2: Perplexity score
==============
WARNING:datasets.builder:Found cached dataset wikitext (/root/.cache/huggingface/datasets/wikitext/wikitext-2-raw-v1/1.0.0/a241db52902eaf2c6aa732210bead40c090019a499ceb13bcbfa3f8ab646a126)
Token indices sequence length is longer than the specified maximum sequence length for this model (287644 > 1024). Running this sequence through the model will result in indexing errors
100% 1120/1124 [02:06<00:00,  8.87it/s]
Perplexity = 24.5659.
```

Stride: 128

```
!python generation.py --ppl

1
Task 2: Perplexity score
==============
WARNING:datasets.builder:Found cached dataset wikitext (/root/.cache/huggingface/datasets/wikitext/wikitext-2-raw-v1/1.0.0/a241db52902eaf2c6aa732210bead40c090019a499ceb13bcbfa3f8ab646a126)
Token indices sequence length is longer than the specified maximum sequence length for this model (287644 > 1024). Running this sequence through the model will result in indexing errors
100% 2240/2248 [04:16<00:00,  8.73it/s]
Perplexity = 24.3746.
```

- As the stride increases the perplexity increases, which means the uncertainty increases and the performance worsens.
- Perplexity in a literal sense means inability to understand or confusion. It is always better to have less confusion i.e, less perplexity. Perplexity can also be defined as a measure of how well a probabilistic model predicts a sample. As the perplexity calculation of sequence generated from the n-gram model or transformer model doesn't change it can be used as a standard measure to compare the performance of models. Its value varies from 1 to infinity. Lower the better.

2. Sentiment Analysis as Text Generation

Task1: Few-shot predictions

1. For the debugging mode and finetuning mode to work I replaced, ‘test_examples’ with test_dir in lines 84 and 101 and replaced and test_doc with doc in line 88 in the sentiment.py file.
2. For the debug mode, !python sentiment.py -d -L 50
3. For the full code, run !python sentiment.py -L 50
4. As the code takes a lot of time to run I have made couple of modifications to run the code fully on a GPU. For that the model and tokenized data are to be loaded on to the GPU.

Results and Analysis:

Debug mode output:

```
!python sentiment.py -d -L 50

positive file no. 996 got: Review: "richard gere can be a commanding actor, but he's ... Sentiment: negative
positive file no. 997 got: Review: "glory--starring matthew broderick, denzel washington, and morgan freeman--is the ... Sentiment: negative
positive file no. 998 got: Review: "steven spielberg's second epic film on world war ii ... Sentiment: negative
Fewshot accuracy: 0.70
```

```
!python sentiment.py -L 50
```

```
positive file no. 905 incorrect
positive file no. 922 incorrect
positive file no. 912 incorrect
positive file no. 980 incorrect
positive file no. 953 incorrect
positive file no. 968 incorrect
positive file no. 952 incorrect
positive file no. 933 incorrect
positive file no. 907 incorrect
positive file no. 944 incorrect
positive file no. 925 incorrect
positive file no. 932 incorrect
positive file no. 977 incorrect
positive file no. 974 incorrect
positive file no. 946 incorrect
positive file no. 924 incorrect
positive file no. 923 incorrect
positive file no. 917 incorrect
positive file no. 976 incorrect
positive file no. 958 incorrect
positive file no. 928 incorrect
positive file no. 926 incorrect
positive file no. 914 incorrect
positive file no. 975 incorrect
positive file no. 964 incorrect
positive file no. 927 incorrect
positive file no. 954 incorrect
positive file no. 981 incorrect
positive file no. 961 incorrect
positive file no. 971 incorrect
positive file no. 901 incorrect
positive file no. 918 incorrect
positive file no. 970 incorrect
positive file no. 939 incorrect
positive file no. 972 incorrect
positive file no. 985 incorrect
positive file no. 990 incorrect
positive file no. 935 incorrect
positive file no. 973 incorrect
positive file no. 996 incorrect
positive file no. 967 incorrect
positive file no. 947 incorrect
positive file no. 991 incorrect
positive file no. 913 incorrect
```

```
positive file no. 963 incorrect
positive file no. 956 incorrect
positive file no. 937 incorrect
positive file no. 909 incorrect
positive file no. 930 incorrect
positive file no. 950 incorrect
positive file no. 911 incorrect
positive file no. 941 incorrect
positive file no. 931 incorrect
positive file no. 949 incorrect
positive file no. 992 incorrect
positive file no. 943 incorrect
positive file no. 942 incorrect
positive file no. 902 incorrect
positive file no. 916 incorrect
positive file no. 920 incorrect
positive file no. 983 incorrect
positive file no. 998 incorrect
positive file no. 978 incorrect
positive file no. 986 incorrect
positive file no. 993 incorrect
positive file no. 966 incorrect
positive file no. 929 incorrect
positive file no. 997 incorrect
positive file no. 936 incorrect
positive file no. 945 incorrect
positive file no. 921 incorrect
positive file no. 989 incorrect
positive file no. 959 incorrect
positive file no. 979 incorrect
positive file no. 955 incorrect
positive file no. 951 incorrect
positive file no. 948 incorrect
positive file no. 919 incorrect
positive file no. 908 incorrect
positive file no. 987 incorrect
positive file no. 915 incorrect
positive file no. 903 incorrect
positive file no. 960 incorrect
positive file no. 994 incorrect
positive file no. 906 incorrect
positive file no. 938 incorrect
positive file no. 962 incorrect
negative file no. 928 incorrect
negative file no. 950 incorrect
negative file no. 962 incorrect
Fewshot accuracy: 0.54
```

- Some changes were made to the training data for the model to predict better
- The model achieved an accuracy of 0.54

Known Issues and Limitations:

- I have looked at the reviews(data), and the following are my observations:
    - The reviews are long and mostly discussed the story
    - The sentiment can only be understood when one reads between the lines
- The performance of model can be attributed to issues with the data.