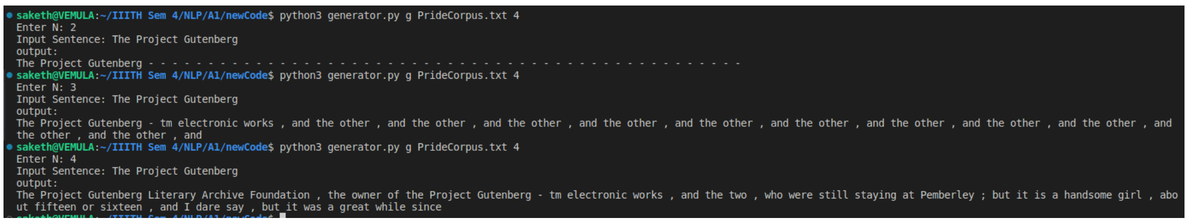
Introduction to NLP
2022114014
Saketh Reddy Vemula
Assignment 1 Report

Analysis of Generator code:

1) Using the generated N-gram models (without the smoothing techniques), try generating sequences. Experiment with different values of N and report which models perform better in terms of fluency.

Output for an input sentence as the value of N increases:



- Enter N: 1
  Input Sentence: Lady Catherine seemed quite astonished at not
  output:
  Lady Catherine seemed quite astonished at not The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The
- Enter N: 2
  Input Sentence: Lady Catherine seemed quite astonished at not
  output:
  Lady Catherine seemed quite astonished at not be so much as to be so much as to be so much as to be so much as to be so much as to be so much as to
- Enter N: 3
  Input Sentence: Lady Catherine seemed quite astonished at not
  output:
  Lady Catherine seemed quite astonished at not having done it for Mr . Darcy
- Enter N: 4
  Input Sentence: Lady Catherine seemed quite astonished at not
  output:
  Lady Catherine seemed quite astonished at not receiving a direct answer ; and , after a short silence , the lady first spoke . The The The The The The The The The The The The The
- Enter N: 5
  Input Sentence: Lady Catherine seemed quite astonished at not
  output:
  Lady Catherine seemed quite astonished at not receiving a direct answer ; and Elizabeth suspected herself to be the first creature who had ever dared to trifle with so much dignified impertinence . The The The The
- Enter N: 6
  Input Sentence: Lady Catherine seemed quite astonished at not
  output:
  Lady Catherine seemed quite astonished at not receiving a direct answer ; and Elizabeth suspected herself to be the first creature who had ever dared to trifle with so much dignified impertinence . The The The The

- Enter N: 7
  Input Sentence: Lady Catherine seemed quite astonished at not
  output:
  Lady Catherine seemed quite astonished at not receiving a direct answer ; and Elizabeth suspected herself to be the first creature who had ever dared to trifle with so much dignified impertinence . The The The The

Analysis:

- As the value of N increases, the quality and flow of the generated sentence improves.
- If we are simply selecting the most probable word at each step greedily, it's quite likely that we'll end up with repetitive sequences, especially if our training data is not diverse enough (which it will never be) or if our model's parameters are not fine-tuned properly (which is very difficult to achieve with certain limitations).
- For smaller N's, the model saturates early because it cannot predict the next word anymore or runs into an infinite loop.
- The sentence saturates as soon as the model cannot predict any word next word or model runs into a loop.
- On increasing N of N-gram upon every word prediction, the model can successfully generate sentences given first few words of that sentence as input sentence. The distance at which it saturates also increases as N increases.

2) Attempt to generate a sentence using an Out-of-Data (OOD) scenario with your N-gram models. Analyze and discuss the behavior of N-gram models in OOD contexts.

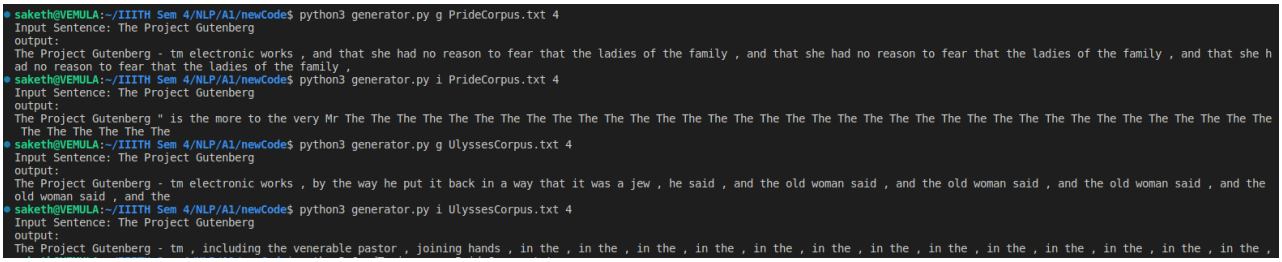
- Enter N: 4
  Input Sentence: admiralty division the summons, exparte
  output:
  admiralty division the summons, exparte The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The The
- Enter N: 2
  Input Sentence: admiralty division the summons, exparte
  output:
  admiralty division the summons, exparte The first , and the room , and the room , and the room , and the room , and the room , and the room , and the room , and the room , and the room , and the room , and the room , and the room , and the room

Analysis:

- The model performs better for small N value as compared to larger ones. For smaller N, the model can at least predict some words. This could be because the word “exparte” happens to be present in training corpus somehow. Otherwise the model fails to predict any words. It gives 0.0 probability to all possible next words and selects whatever word is first in sorted Counter object.
- Model fails miserably in OOD scenarios.

3) Now try to generate text using the models with the smoothing techniques (LM1, LM2, LM3, LM4).

LM1, LM2, LM3, and LM4:



Analysis:

- All the language models runs into loop after certain length.
- Sometimes they can’t predict the next word. They give 0.0 probability to all possible words (all unigrams).
- Picking most probable word at each step greedily, it’s quite likely that we’ll end up with repetitive sequences, especially if our training data is not diverse enough (which it will never be) or if our model’s parameters are not fine-tuned properly (which is very difficult to achieve with certain limitations).
- Generally, for most of the sentences, as N increases the fluency of the model gets better.