Foundations of AI
Yunyu Guo
Assignment 1

- Describe how the choice of training dataset affected your model. What happens if we train a model on one type of data (like music lyrics) and then ask it to work with a different type of data (such as medical reports)? Which datasets do popular language models use, and how might this affect their output? (1 paragraph)

  I use medical record as a test of the model, when I use prompts already in the medical record such as “patient” and “physical”, the model seems to generate the rest context more relevant to the medical record. When I use prompt that is not in the medical record, such as “for information on our return policy”, the generated context seems just concatenate random characters. So the choice of training dataset affects the performance and behaviour of a language model.
  Models trained on specific types of data learn patterns, vocabulary, and unique to that dataset. Common Datasets: Wikipedia, OpenWebText: A dataset derived from web pages. Effect on Output: Models trained on diverse datasets can handle a wide range of topics. The content and quality of the training data can lead to biases affecting the model's output. (see screenshots of n =3, n=5, n=20 at the end)

- How long did this assignment take you? (1 sentence)
  1 day

- Whom did you work with, and how? (1 sentence each)
  - Discussing the assignment with others is encouraged, as long as you don’t share the code.
    I discuss the outcomes with other student to verify if using relevant and irrelevant prompts would affect the model output.

- Which resources did you use? (1 sentence each)
  - For each, please list the URL and a brief description of how it was useful.

  Statistical Language Model: N-gram to calculate the Probability of word sequence using Python: https://medium.com/codex/statistical-language-model-n-gram-to-calculate-the-probability-of-word-sequence-using-python-2e54a1084250

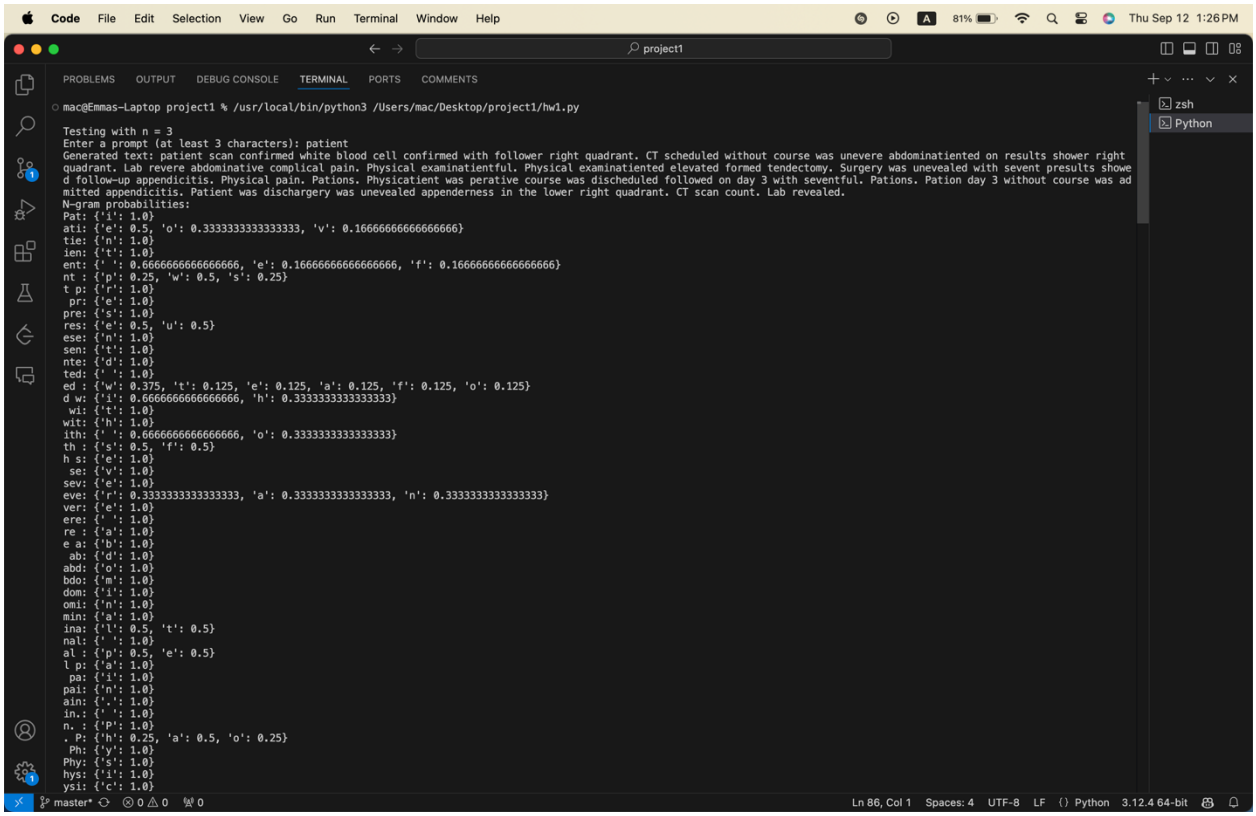
- A few sentences about:
  - What was the most difficult part of the assignment?
    How to store the n-gram and its probability respectively. What data structure should I use.

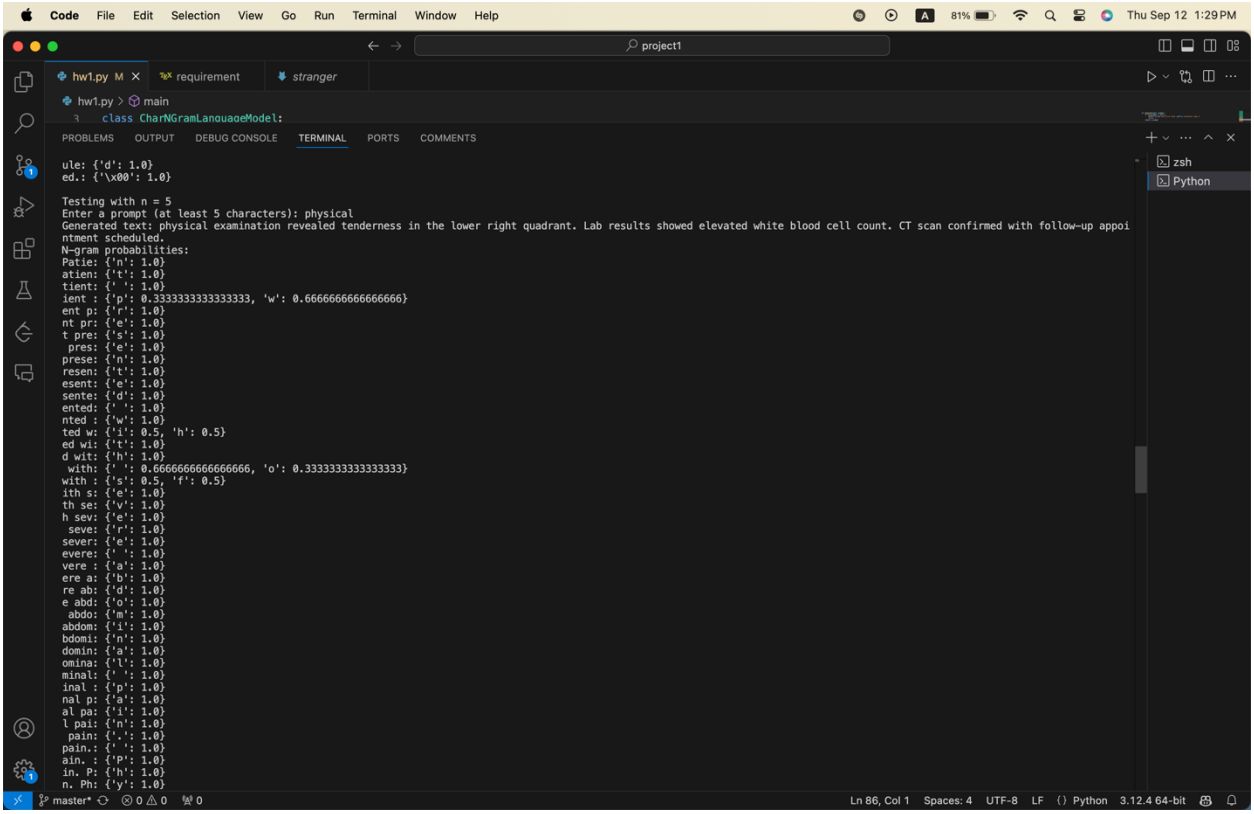
- What was the most rewarding part of the assignment?
  To be able to generate content after enter the prompt and to design a model that can generate the next character based on probability

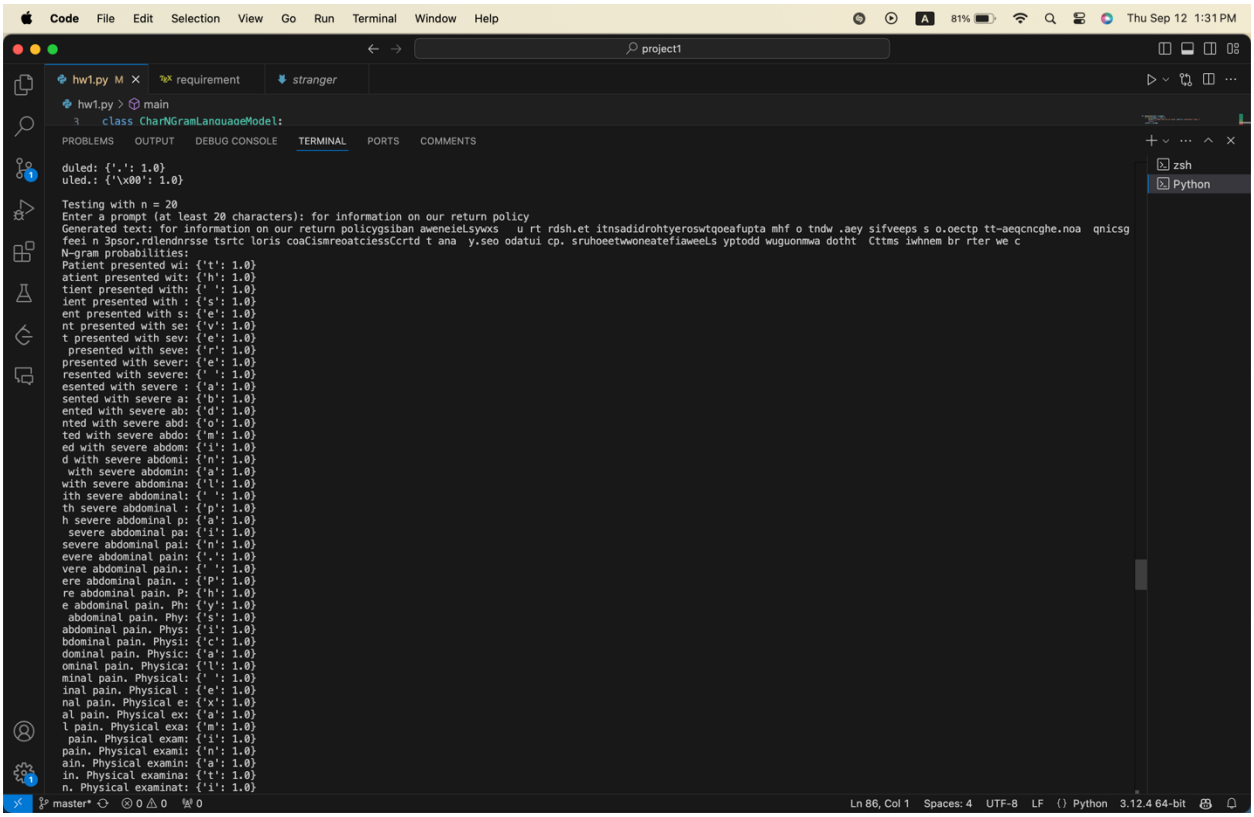
- What did you learn doing the assignment?
  How to build a nested dictionary.

- Constructive and actionable suggestions for improving assignments, office hours, and class time are always welcome.

  Could you provide relevant reading materials to every assignment? For example, in this assignment, provide options of what data structure we should use, the links to materials of how to construct nested dictionary, how a general n-gram model is created with some coding samples.




Code File Edit Selection View Go Run Terminal Window Help
Thu Sep 12 1:29 PM
project1
hw1.py M
requirement
stranger
hw1.py > main
class CharNGramLanguageModel:
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
ule: {'d': 1.0}
ed.: {'\x00': 1.0}
Testing with n = 5
Enter a prompt (at least 5 characters): physical
Generated text: physical examination revealed tenderness in the lower right quadrant. Lab results showed elevated white blood cell count. CT scan confirmed with follow-up appoi
ntment scheduled.
N-gram probabilities:
Patie: {'n': 1.0}
atien: {'t': 1.0}
tient: {' ': 1.0}
ient : {'p': 0.3333333333333333, 'w': 0.6666666666666666}
ent p: {'r': 1.0}
nt pr: {'e': 1.0}
t pre: {'s': 1.0}
pres: {'e': 1.0}
prese: {'n': 1.0}
resen: {'t': 1.0}
esent: {'e': 1.0}
sente: {'d': 1.0}
ented: {' ': 1.0}
nted : {'w': 1.0}
ted w: {'i': 0.5, 'h': 0.5}
ed wi: {'t': 1.0}
d wit: {'h': 1.0}
with: {' ': 0.6666666666666666, 'o': 0.3333333333333333}
with : {'s': 0.5, 'f': 0.5}
ith s: {'e': 1.0}
th se: {'v': 1.0}
h sev: {'e': 1.0}
seve: {'r': 1.0}
sever: {'e': 1.0}
severe: {' ': 1.0}
vere : {'a': 1.0}
ere a: {'b': 1.0}
re ab: {'d': 1.0}
e abd: {'o': 1.0}
abdo: {'m': 1.0}
abdom: {'i': 1.0}
bdomi: {'n': 1.0}
domin: {'a': 1.0}
omina: {'l': 1.0}
minal: {' ': 1.0}
inal : {'p': 1.0}
nal p: {'a': 1.0}
al pa: {'i': 1.0}
l pai: {'n': 1.0}
pain: {'.': 1.0}
pain.: {' ': 1.0}
ain. : {'P': 1.0}
in. P: {'h': 1.0}
n. Ph: {'y': 1.0}
master*
Ln 86, Col 1 Spaces: 4 UTF-8 LF Python 3.12.4 64-bit
zsh
Python


Code File Edit Selection View Go Run Terminal Window Help
Thu Sep 12 1:31 PM
project1
hw1.py M
requirement
stranger
hw1.py > main
class CharNGramLanguageModel:
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS
duled: {'.': 1.0}
uled.: {'\x00': 1.0}
Testing with n = 20
Enter a prompt (at least 20 characters): for information on our return policy
Generated text: for information on our return policygsiban aweneieLsywxs u rt rdsh.et itnsadidrohtyeroswtqoeafupta mhf o tndw .aey sifveeps s o.oectp tt-aeqcncghe.noa qnicsg
feei n 3psor.rdlendnrsse tsrtc loris coaCismreoatciessCcrtd t ana y.seo odatui cp. sruhoeetwwoneatefiaweeLs yptodd wuguonmwa dotht Cttms iwhnem br rter we c
N-gram probabilities:
Patient presented wi: {'t': 1.0}
atient presented wit: {'h': 1.0}
tient presented with: {' ': 1.0}
ient presented with : {'s': 1.0}
ent presented with s: {'e': 1.0}
nt presented with se: {'v': 1.0}
t presented with sev: {'e': 1.0}
presented with seve: {'r': 1.0}
presented with sever: {'e': 1.0}
resented with severe: {' ': 1.0}
esented with severe : {'a': 1.0}
sented with severe a: {'b': 1.0}
ented with severe ab: {'d': 1.0}
nted with severe abd: {'o': 1.0}
ted with severe abdo: {'m': 1.0}
ed with severe abdom: {'i': 1.0}
d with severe abdomi: {'n': 1.0}
with severe abdomin: {'a': 1.0}
with severe abdomina: {'l': 1.0}
ith severe abdominal: {' ': 1.0}
th severe abdominal : {'p': 1.0}
h severe abdominal p: {'a': 1.0}
severe abdominal pa: {'i': 1.0}
severe abdominal pai: {'n': 1.0}
evere abdominal pain: {'.': 1.0}
vere abdominal pain.: {' ': 1.0}
ere abdominal pain. : {'P': 1.0}
re abdominal pain. P: {'h': 1.0}
e abdominal pain. Ph: {'y': 1.0}
abdominal pain. Phy: {'s': 1.0}
abdominal pain. Phys: {'i': 1.0}
bdominal pain. Physi: {'c': 1.0}
dominal pain. Physic: {'a': 1.0}
ominal pain. Physica: {'l': 1.0}
minal pain. Physical: {' ': 1.0}
inal pain. Physical : {'e': 1.0}
nal pain. Physical e: {'x': 1.0}
al pain. Physical ex: {'a': 1.0}
l pain. Physical exa: {'m': 1.0}
pain. Physical exam: {'i': 1.0}
ain. Physical exami: {'n': 1.0}
in. Physical examin: {'a': 1.0}
n. Physical examina: {'t': 1.0}
. Physical examinat: {'i': 1.0}
master*
Ln 86, Col 1 Spaces: 4 UTF-8 LF Python 3.12.4 64-bit
zsh
Python