N-gram Language Model

An N-gram language model is a statistical approach used in Natural Language Processing to predict the next word or character in a sequence based on the previous *(n−1)* tokens. It learns patterns by counting the frequency of token sequences in a given text corpus. N-gram models are simple, interpretable, and commonly used as baseline models for text generation, but they are limited by their fixed context size.

```
import random
from collections import defaultdict
text = """
artificial intelligence is transforming modern society.
it is used in healthcare finance education and transportation.
machine learning allows systems to improve automatically with experience.
data plays a critical role in training intelligent systems.
large datasets help models learn complex patterns.
deep learning uses multi layer neural networks.
neural networks are inspired by biological neurons.
each neuron processes input and produces an output.
training a neural network requires optimization techniques.
gradient descent minimizes the loss function.

natural language processing helps computers understand human language.
text generation is a key task in nlp.
language models predict the next word or character.
recurrent neural networks handle sequential data.
lstm and gru models address long term dependency problems.
however rnn based models are slow for long sequences.

transformer models changed the field of nlp.
they rely on self attention mechanisms.
attention allows the model to focus on relevant context.
transformers process data in parallel.
this makes training faster and more efficient.
modern language models are based on transformers.

education is being improved using artificial intelligence.
intelligent tutoring systems personalize learning.
automated grading saves time for teachers.
online education platforms use recommendation systems.
technology enhances the quality of learning experiences.

ethical considerations are important in artificial intelligence.
fairness transparency and accountability must be ensured.
ai systems should be designed responsibly.
data privacy and security are major concerns.
researchers continue to improve ai safety.

text generation models can create stories poems and articles.
they are used in chatbots virtual assistants and content creation.
generated text should be meaningful and coherent.
evaluation of text generation is challenging.
human judgement is often required.

continuous learning is essential in the field of ai.
research and innovation drive technological progress.
students should build strong foundations in mathematics.
programming skills are important for ai engineers.
practical experimentation enhances understanding.
"""


text = text.lower().replace("\n", " ")

# -------------------------
# BUILD BIGRAM MODEL
# -------------------------
n = 2  # Bigram
ngram_model = defaultdict(list)

for i in range(len(text) - n + 1):
    prefix = text[i:i+n-1]#extracts first n word/tokens
    next_char = text[i+n-1]#extracts the next possible charcaters
    ngram_model[prefix].append(next_char)

# -------------------------
# TEXT GENERATION FUNCTION
# -------------------------
def generate_text(seed, length=300):
```

```
def generate_text(seed, length=300):
    output = seed
    for _ in range(length):
        prefix = output[-(n-1):]#extracts prefix
        next_char = random.choice(ngram_model.get(prefix, [' ']))#randomly selects next possible characters
        output += next_char
    return output

# -------------------------
# GENERATED OUTPUT
# -------------------------
seed_text = "is"
generated_text = generate_text(seed_text)

print("Generated Text using N-gram Model:\n")
print(generated_text)
```

```
Generated Text using N-gram Model:

isfoforechiond iniomope. monge n. ioloud d ndexprngeunen cl hoh poplevatourkiacintalpompllin inedimsel in cauten. arng finti
```

Limitations of N-gram Model

- **Limited Context:**
  N-gram models consider only a fixed number of previous tokens, so they cannot capture long-term dependencies in text.
- **No Semantic Understanding:**
  The model is purely statistical and does not understand the meaning of words or sentences.
- **Data Sparsity:**
  For larger values of n, many n-grams do not appear in the training data, leading to poor predictions.
- **Poor Text Coherence:**
  Generated text often lacks grammatical structure and becomes incoherent for longer sequences.
- **Not Scalable:**
  N-gram models require large memory and perform poorly on complex, real-world NLP tasks.