Observations from the Program Execution

After running the script, you may observe the following outcomes based on different scenarios:

1. Successful Execution:

✅ **If the word exists in the dataset and has associated bigrams**, the program generates a meaningful sentence.

Example Output:

Asia is a major economy in the world.

Here, the program successfully finds bigrams associated with "Asia" and randomly selects words to form a sentence.

2. Word Not Found in Training Data:

⚠️ **If the input word is not present in the dataset,** the program displays:

'Asia' not found in training data.

This indicates that "Asia" was not found in any bigrams, meaning the dataset should be expanded.

Possible Causes & Fixes:

- The dataset is too small or doesn't contain "Asia".
- The text preprocessing removed "Asia" due to special characters or inconsistencies.
- Consider adding more text files with diverse vocabulary.

3. Short or Incomplete Sentences:

⚠️ **If a word has very few associated bigrams,** the output might be too short or incomplete.

Example Output:

Asia is

This happens when "Asia" appears in the dataset but has only one or two known bigrams, making it impossible to continue the sentence further.

Possible Fixes:

- Increase the dataset size.
- Reduce topk from 3 to 2 to include less frequent bigrams.

4. Repetitive or Nonsensical Output:

⚠️ **If bigrams are not diverse,** the sentence may become repetitive or nonsensical.

Example Output:

Asia Asia Asia Asia is is is a a a country.

This happens if certain bigrams are **too frequent** in the dataset, leading the program to select the same words repeatedly.

Possible Fixes:

- Implement **weighted selection** instead of pure randomness.
- Expand dataset diversity to include richer vocabulary.

5. Performance Considerations:

⏳ **If the dataset is large,** the script may take longer to process.

- Computing bigrams and their frequency requires extra memory and CPU time.
- Consider **limiting the dataset size** or optimizing topk values.

Summary of Observations

Scenario	Observation	Fix / Improvement
✅ Successful Execution	Generates meaningful sentences.	No fix needed.
⚠️ Word Not Found	No sentence is generated.	Add more text data.
⚠️ Short Sentence	Sentence stops too soon.	Expand dataset, reduce topk.
⚠️ Repetitive/Nonsensical Output	Loops words awkwardly.	Use weighted selection, diversify dataset.
⏳ Slow Execution	Large datasets take time.	Optimize processing, use smaller files.

Next Steps

- Add **more varied training data** for better sentence diversity.
- Implement **probabilistic word selection** for more natural text generation.
- Explore **trigram or higher-order n-grams** for improved sentence coherence.