Assignment 6

 # Question 1:
What is an LLM? How does it relate/differ from n-gram models? [2 points]
 # Solution 1:
LLM (Large Language Model):
An LLM, or Large Language Model, is a type of artificial intelligence model designed to understand and generate human language. It is based on deep learning architectures, particularly neural networks, and is trained on massive amounts of text data to predict and produce coherent and contextually appropriate text. Examples of LLMs include OpenAI's GPT-3 and GPT-4, Google's BERT, and others. These models leverage techniques like transformers, attention mechanisms, and vast training datasets to capture the complexities of language.

Relation/Difference:

- Contextual Understanding:
    - LLMs: Understand long-range dependencies and context.
    - n-gram Models: Limited to short-range context of $n-1$ words.
- Flexibility:
    - LLMs: Versatile, used in various applications like translation and summarization.
    - n-gram Models: Simpler, often used for basic text generation.

 # Question 2:
What is template-based learning? What's its advantage? Give one example. [2 points]
 # Solution 2:
Template-based learning involves the use of predefined templates to guide the generation or interpretation of text. These templates can include fixed sentence structures with slots for variable insertion, allowing the system to produce consistent and structured outputs.

Advantage:
The primary advantage of template-based learning is its ability to ensure grammatical correctness and maintain a consistent structure in generated text. This approach is particularly useful for applications where reliability and clarity are paramount, as it reduces the risk of producing incoherent or contextually incorrect sentences.

Example:
One common example of template-based learning is in automated customer service chatbots. For instance, a template might be used to generate responses to frequently asked questions:

Template: "Hello, [Customer Name]. Thank you for contacting us about [Issue]. We are currently working on resolving this issue and will get back to you shortly."

By filling in the variables (customer name and issue), the system can generate polite, contextually appropriate responses while ensuring grammatical correctness and a professional tone. This helps maintain a high standard of communication quality in customer service interactions.

 # Question 3:
We are all about different types of experimentations for this assignment question, but it's up to you what those experiments look like! Expand and modify the class code at least three different ways and report your results for loss (quantitative) and generation (qualitative). For example, you can train a model using a different (ideally, larger) dataset, change some of the hyper-parameters (block size, batch size, iterations, etc.), extend bigram model to trigram, etc. There are no right or wrong answers here and you shouldn't shy away from trying something interesting (and not just something easy). Present your experiments with description (e.g., what did you change or do differently), results (training/validation performance and generation quality), and your thoughts (1-2 sentences) about what happened here. [6 points]
 # Solution 3:

- You can change *number of iterations*
- Change *Hyperparameters* like *Batch size* (32 to 8 for example) or *Block size* (8 to 16 eg)
- *Change architecture*, eg from Bigram to Trigram
- *Modify the dataset* and make it larger / more suitable