Problem Statement: You are given a dataset of questions related to AI/ML. Your objective is to:

1. Generate new questions based on the dataset by slightly modifying the given ones while ensuring they remain relevant and coherent.

2. Maintain semantic consistency while altering phrasing, structure, or complexity.

3. Ensure varying difficulty levels in the generated questions to cover basic, intermediate, and advanced AI/ML topics.

Solution:

1. First we find the solution to these questions, i am pretty sure for the real internship dataset the answers will be given to us but for this task let's assume we don't have the answers, so using open ai's api we generate the answers in a particular format which properly divides the answers into steps, and infer if its calculative or conceptual
2. Next step is to use Term Frequency - Inverse Document Frequency (TF-IDF) to convert the text into numerical data. It gives more weightage to important words and less to the common words like 'is', 'the', etc..
3. Only going through the text isn't enough to get a numeric representation so we also go through the answers and add numeric weightage to each question based on steps taken, calculative or conceptual etc..
4. Classify each question as Easy, medium or Hard using K Mean Clustering as we now have a mathematical representation of the questions/dataset
5. Classify each question based on which topic it's related to, but for this task we can ignore this step
6. Then we use open ai's api or hugging face transformers to change the difficulty of each question by one notch(Easy->Medium, Medium->Hard) while retaining the original question, if the question is already hard we just rephrase it.

I have some test code for the classification part, i tried to use open ai's api but was repeatedly running into rate limit error, might need some credits to use that, but i am pretty sure it should work

Repo link to the file:
https://github.com/Sanchit9587/Genstrive-Question-Generation/blob/main/Testing.ipynb

This method is efficient, will take less time to build and quite repeatable. The advantage of this method is that the data set gets larger and gets classified at the same time as we know the labels for the newer questions and makes it easier to work with in the future.