GENERATIVE AI AND ITS APPLICATIONS

HANDS – ON 1

Name: SRIVIDHYA M

SRN: PES2UG23CS602

Section: J

Email ID: srividyavidya022@gmail.com

Mobile Number:8073960843

Github Link: [s-r-i-v-i-d-h-y-a-m/PES2UG23CS602_GENAI_Hands-on-1]

Assignment 1: screenshot

Task	**Model**	**Classification (Success/Failure)**	**Observation (What actually happened?)**	**Why did this happen? (Architectural Reason)**
Generation	BERT	Failure	Generated repetitive symbols instead of meaningful text	BERT is encoder-only and is trained for understanding text, not autoregressive generation, so it cannot predict text step-by-step.
Generation	RoBERTa	Failure	Returned only the input prompt without adding new text	RoBERTa is also encoder-only, meaning it has no decoder to generate new sequences.
Generation	BART	Partial Success	Generated new text but it was noisy and incoherent	BART has an encoder–decoder architecture, so it *can* generate text, but if not fine-tuned for open-ended generation, output may be low quality.
Fill-Mask	BERT	Success	Predicted “create”, “generate”, “produce” correctly	BERT is trained using Masked Language Modeling (MLM), so it predicts missing words effectively.

				effectively.
Fill-Mask	RoBERTa	Success	Predicted “generate”, “create” with high confidence	RoBERTa is an optimized version of BERT, trained extensively on MLM, which makes fill-in predictions very accurate.
Fill-Mask	BART	Partial Success	Predicted “create” as top choice, followed by “help”, “provide”, “enable”, “improve”; predictions make sense but confidence is low	BART is an encoder–decoder model, capable of MLM, but it is primarily optimized for sequence-to-sequence tasks, not specifically masked word prediction.
Question Answering	BERT	Partial Success	Predicted “Generative AI poses” with low confidence (score ~0.006); model weights for qa_outputs were newly initialized, so fine-	BERT is pre-trained for MLM and next sentence prediction, but QA requires fine-tuning on a downstream SQuAD-like task; untrained
			tuning is required for accurate results	qa_outputs weights lead to poor predictions
Question Answering	Roberta	Partial Success	Predicted “Generative AI poses significant risks such as hallucinations, bias, and” with low confidence (score ~0.008); QA head weights were newly initialized	RoBERTa is pre-trained as an encoder-only MLM; QA requires fine-tuning on a downstream dataset; uninitialized qa_outputs weights lead to poor predictions
Question Answering	Bart	Partial Success	Predicted “Generative” with low confidence (score ~0.043); QA head weights (qa_outputs) were newly initialized, so fine-tuning is needed	BART has an encoder–decoder architecture and can perform QA, but without fine-tuning on a QA dataset, predictions are very limited and short.

Assignment 2: Screenshots

Study Buddy (PDF Quizzer)

- **Goal**: Paste a textbook chapter (Context) and ask "What is the definition of X?".
- **Tech**: Extractive QA pipeline.

```
!pip install transformers torch PyPDF2
```

Show hidden output

```
from transformers import pipeline
import PyPDF2


qa_pipeline = pipeline(
    "question-answering",
    model="distilbert-base-cased-distilled-squad"
)

print("STUDY BUDDY - PDF QUIZZER")



file_path = "2.pdf"


context = ""

with open(file_path, "rb") as file:
```

[9]

```python
with open(file_path, "rb") as file:
    reader = PyPDF2.PdfReader(file)
    for page in reader.pages:
        text = page.extract_text()
        if text:
            context += text

print("PDF loaded successfully!")
print(f" Total characters extracted: {len(context)}")


question = "What is the objective  of Unsupervised learning?"

result = qa_pipeline(
    question=question,
    context=context
)

print("\n Question:")
print(question)

print("\n Answer:")
print(result["answer"])
```

```
Device set to use cpu
STUDY BUDDY - PDF QUIZZER
PDF loaded successfully!
 Total characters extracted: 1884

 Question:
What is the objective  of Unsupervised learning?

 Answer:
to identify hidden
patterns   within the data without any predefined labels
```

Text Generation (BERT, RoBERTa, BART)

Key concepts understood

Text generation requires autoregressive decoding, where the model predicts the next token step by step. Encoder-only models such as BERT and RoBERTa are designed mainly for language understanding tasks and do not support true text generation. Encoder–decoder models like BART include a decoder, which allows them to generate text, but the quality of generation depends on proper fine-tuning for open-ended tasks. Model architecture plays a critical role in determining whether a task can be performed successfully.

The solution implemented

Text generation was attempted using BERT, RoBERTa, and BART models. BERT produced repetitive or meaningless symbols due to its encoder-only architecture. RoBERTa returned only the input prompt without generating new content. BART generated new text, but the output was noisy and partially incoherent because it was not fine-tuned for open-ended generation. Based on these observations, BERT and RoBERTa were classified as failures, while BART was classified as partial success.

Fill-Mask Task (BERT, RoBERTa, BART)

Key concepts understood

Masked Language Modeling involves predicting missing words using surrounding context. BERT and RoBERTa are extensively trained on MLM objectives, making them well suited for fill-mask tasks. RoBERTa improves upon BERT by using larger datasets and optimized training strategies. Although BART can perform masked prediction, it is primarily optimized for sequence-to-sequence tasks rather than direct masked word prediction.

The solution implemented

A fill-mask pipeline was applied to sentences containing masked tokens. BERT accurately predicted appropriate words such as create, generate, and produce. RoBERTa predicted similar words with higher confidence scores. BART predicted reasonable alternatives, but with lower confidence. As a result, BERT and RoBERTa were marked as successful, while BART showed partial success. This confirmed that MLM-trained models perform best for fill-mask tasks.

Question Answering (BERT, RoBERTa, BART)

Key concepts understood

Extractive question answering involves selecting the correct answer span directly from the given context. QA models require fine-tuning on datasets such as SQuAD to achieve good accuracy. Without fine-tuning, QA heads remain uninitialized, leading to low-confidence and incomplete predictions. Both encoder-only and encoder–decoder models can perform QA, but training plays a major role in output quality.

The solution implemented

Question answering pipelines were tested using BERT, RoBERTa, and BART models. All models produced short answers with low confidence scores. System warnings indicated that QA output layers were newly initialized and not fine-tuned. Therefore, all models were classified as partial success. The experiment demonstrated the importance of downstream task-specific fine-tuning.

Study Buddy (PDF Quizzer)

Key concepts understood
Extractive question answering can be used to build document-based quiz systems. The quality of answers depends on the relevance and completeness of the provided context. Pre-trained QA models fine-tuned on SQuAD can

effectively extract answers from long passages. PDF text extraction is required to convert unstructured documents into usable text for NLP tasks.

The solution implemented

A Study Buddy PDF Quizzer was implemented using an extractive question answering pipeline. Text was extracted from a textbook PDF using PyPDF2 and combined into a single context string. A question was provided by the user, and the context along with the question was passed to a DistilBERT model fine-tuned on SQuAD. The model successfully extracted accurate answers directly from the PDF content, demonstrating effective document-based question answering.