

# Generative Ai

## Unit 1 Assignment – Model Benchmark

Challenge

Name :Amar

SRN:PES2UG23CS052

Section: A

Date:25-01-2026

Output Screenshots:

Experiment-1 : Text Generation

Code:

```
prompt = "The future of Artificial Intelligence is"  
for name, model in models.items():  
    print(f"\n{name} Output:")  
    try:  
        generator = pipeline("text-generation", model=model)  
        print(generator(prompt, max_length=30))  
    except Exception as e:  
        print("Failed:", e)
```

# Output Screenshot:

```
BERT Output:  
Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better performance, install the package with: `pip install hf_xet`  
If you want to use 'BertLMHeadModel' as a standalone, add `is_decoder=True`.  
Device set to use cpu  
Truncation was not explicitly activated but `max_length` is provided a specific value, please use `truncation=True` to explicitly truncate examples to max length. Defaulting to 'longest'.  
Both `max_new_tokens` (-256) and `max_length` (=30) seem to have been set. `max_new_tokens` will take precedence. Please refer to the documentation for more information. (https://huggingface.co/docs/transformers/main\_classes/text\_generation#generating-predictions)  
[{"generated_text": "The future of Artificial Intelligence is....."}]  
  
RoBERTA Output:  
c:\Users\Amar_Sagar\AppData\Local\Programs\Python\Python312\Lib\site-packages\huggingface_hub\file_download.py:143: UserWarning: 'huggingface_hub' cache-system uses symlinks by default  
To support symlinks on Windows, you either need to activate Developer Mode or to run Python as an administrator. In order to activate developer mode, see this article: https://docs.microsoft.com/en-us/windows/uwp/debug-test-perf/enable-developer-mode  
warnings.warn(message)  
Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better performance, install the package with: `pip install hf_xet`  
If you want to use 'RobertaLMHeadModel' as a standalone, add `is_decoder=True`.  
Device set to use cpu  
Truncation was not explicitly activated but `max_length` is provided a specific value, please use `truncation=True` to explicitly truncate examples to max length. Defaulting to 'longest'.  
Both `max_new_tokens` (-256) and `max_length` (=30) seem to have been set. `max_new_tokens` will take precedence. Please refer to the documentation for more information. (https://huggingface.co/docs/transformers/main\_classes/text\_generation#generating-predictions)  
[{"generated_text": "The future of Artificial Intelligence is....."}]  
  
BART Output:  
c:\Users\Amar_Sagar\AppData\Local\Programs\Python\Python312\Lib\site-packages\huggingface_hub\file_download.py:143: UserWarning: 'huggingface_hub' cache-system uses symlinks by default  
To support symlinks on Windows, you either need to activate Developer Mode or to run Python as an administrator. In order to activate developer mode, see this article: https://docs.microsoft.com/en-us/windows/uwp/debug-test-perf/enable-developer-mode  
warnings.warn(message)  
Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better performance, install the package with: `pip install hf_xet`  
Some weights of BartForCausalLM were not initialized from the model checkpoint at facebook/bart-base and are newly initialized: ['lm_head.weight', 'model.decoder.embed_tokens.weight']  
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.  
Device set to use cpu  
Truncation was not explicitly activated but `max_length` is provided a specific value, please use `truncation=True` to explicitly truncate examples to max length. Defaulting to 'longest'.  
Both `max_new_tokens` (-256) and `max_length` (=30) seem to have been set. `max_new_tokens` will take precedence. Please refer to the documentation for more information. (https://huggingface.co/docs/transformers/main\_classes/text\_generation#generating-predictions)  
[{"generated_text": "The future of Artificial Intelligence is....."}]
```

# Experiment-2 : Masked Language Modeling (Fill Mask)

## Code:

```
for name, model in models.items():  
  
    print(f"\n{name} Output:")  
  
    try:  
  
        masker = pipeline("fill-mask", model=model)  
  
        result = masker(sentences[name])  
  
        print(result[:2]) # show top 2 predictions  
  
    except Exception as e:  
  
        print("Failed:", e)
```

# Output Screenshot:

```
BERT Output:  
Some weights of the model checkpoint at bert-base-uncased were not used when initializing BertForMaskedLM: ['bert.pooler.dense.bias', 'bert.pooler.dense.weight', 'cls.seq_relationship'.  
- This IS expected if you are initializing BertForMaskedLM from the checkpoint of a model trained on another task or with another architecture (e.g. initializing a BertForSequenceClassification model from the checkpoint of a BertForMaskedLM model).  
- This IS NOT expected if you are initializing BertForMaskedLM from the checkpoint of a model that you expect to be exactly identical (initializing a BertForSequenceClassification model from the checkpoint of a BertForMaskedLM model).  
Device set to use cpu  
[{"score": 0.539692759513855, "token": 3443, "token_str": "create", "sequence": "the goal of generative ai is to create new content."}, {"score": 0.15575766563415527, "token": 9699, "token_str": "content", "sequence": "the goal of generative ai is to create new content."}]  
  
RoBERTA Output:  
Device set to use cpu  
[{"score": 0.37113118171691895, "token": 5368, "token_str": " generate", "sequence": "The goal of Generative AI is to generate new content."}, {"score": 0.3677138090133667, "token": 1604, "token_str": "content", "sequence": "The goal of Generative AI is to generate new content."}]  
  
BART Output:  
Device set to use cpu  
[{"score": 0.07461544126272202, "token": 1045, "token_str": " create", "sequence": "The goal of Generative AI is to create new content."}, {"score": 0.06571853160858154, "token": 244, "token_str": "content", "sequence": "The goal of Generative AI is to create new content."}]
```

# Experiment-3 :Question Answering

```
question = "What are the risks?"  
  
context = "Generative AI poses significant risks such as hallucinations, bias, and deepfakes."  
  
for name, model in models.items():  
  
    print(f"\n{name} Output:")  
  
    try:  
  
        qa = pipeline("question-answering", model=model)  
  
        print(qa(question=question, context=context))  
  
    except Exception as e:  
  
        print("Failed:", e)
```

## Output Screenshot:

```
BERT Output:  
Some weights of BertForQuestionAnswering were not initialized from the model checkpoint at bert-base-uncased and are newly initialized: ['qa_outputs.bias', 'qa_outputs.weight']  
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.  
Device set to use cpu  
{'score': 0.007576475851237774, 'start': 72, 'end': 82, 'answer': 'deepfakes.'}  
  
ROBERTA Output:  
Some weights of RobertaForQuestionAnswering were not initialized from the model checkpoint at roberta-base and are newly initialized: ['qa_outputs.bias', 'qa_outputs.weight']  
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.  
Device set to use cpu  
{'score': 0.008387230336666107, 'start': 68, 'end': 81, 'answer': 'and deepfakes'}  
  
BART Output:  
Some weights of BartForQuestionAnswering were not initialized from the model checkpoint at facebook/bart-base and are newly initialized: ['qa_outputs.bias', 'qa_outputs.weight']  
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.  
Device set to use cpu  
{'score': 0.041007031686660402, 'start': 72, 'end': 81, 'answer': 'deepfakes'}
```

## Observation table:

Task	Model	Success/Failure	Observation	Architectural Reason
Generation	BERT	Failure	Could not generate text	Encoder-only
Generation	RoBERTa	Failure	Similar to BERT	Encoder-only
Generation	BART	Success	Generated coherent text	Encoder-Decoder
Fill-Mask	BERT	Success	Correct predictions	MLM training
Fill-Mask	RoBERTa	Success	Accurate predictions	Optimized MLM
Fill-Mask	BART	Partial	Less accurate	Not MLM-based
QA	BERT	Partial	Inconsistent answers	Not QA fine-tuned
QA	RoBERTa	Partial	Similar behavior	Base model
QA	BART	Partial	Random answers	Not QA fine-tuned