GenAI HandsOn 1

Name: Arnav Sinha
SRN: PES2UG23CS092
Sec: B

Repo: https://github.com/Piyush-sahoo/GenAI-Hands-on

Observation Table:

Task	Model	Result	Observation	Why it Happened
Generation	BERT	Not Supported	Generation not supported due to encoder-only architecture.	Encoder-only architecture; lacks decoder for autoregressive generation.
Generation	RoBERTa	Not Supported	Generation not supported due to encoder-only architecture.	Encoder-only model; optimized for understanding, not text generation.
Generation	BART	Success	Generated a text sequence using seq2seq decoding.	Encoder–decoder architecture; explicitly trained for seq2seq generation.
Fill-Mask	BERT	Success	Predicted masked token candidates with confidence scores.	Pretrained with Masked Language Modeling objective; strong predictions.
Fill-Mask	RoBERTa	Success	Predicted masked token candidates with confidence scores.	Improved MLM training with dynamic masking and larger corpus.
Fill-Mask	BART	Partial Success	Predicted masked token candidates with confidence scores.	Denoising autoencoder pretraining; MLM not its primary objective.
QA	BERT	Success	Produced an answer span with associated confidence.	Base model without QA fine-tuning; extraction quality is limited.
QA	RoBERTa	Success	Produced an answer span with associated confidence.	Base model not trained on QA datasets; answers may be inaccurate.
QA	BART	Success	Produced an answer span with associated confidence.	Seq2seq base model; not optimized for extractive QA tasks.

1. Original version:
   The original version of a model is the full-sized, fully trained neural network with all its parameters intact. It generally offers higher accuracy and richer representations, but requires more memory, computation, and inference time.

2. Distilled version:
   A distilled version is a compressed and optimized model created using knowledge distillation, where a smaller *student* model learns to mimic the behavior of a larger *teacher* model. It retains most of the original model's performance while being faster, lighter, and more resource-efficient.

3. Seed ()

A seed is an initial numerical value used by a pseudo-random number generator to determine the starting point of a random sequence. Setting a seed ensures that the sequence of "random" values produced is deterministic and reproducible.

Seed (42) and seed (200):
seed (42) and seed (200) are simply two different seed values. Each initializes the random number generator at a different starting state, resulting in **different random sequences**, while still guaranteeing that the same sequence is produced every time the code is run with that seed.

In essence:

- Same seed → same results every run
- Different seeds → different (but repeatable) results

4. Temperature:
   Temperature is a hyperparameter used in text generation that controls the randomness of model predictions. It scales the probability distribution over possible next tokens.

- Low temperature (< 1.0): The model becomes more confident and deterministic, repeatedly choosing high-probability tokens.
- High temperature (> 1.0): The model becomes more creative and diverse, allowing lower-probability tokens to be selected more often.

5. Tokenization:
   Tokenization is the process of converting raw text into smaller units called tokens (such as words, subwords, or characters) that a language model can understand. These tokens are then mapped to numerical IDs before being processed by the model.
   Modern NLP models typically use subword tokenization, which helps handle rare words, spelling variations, and large vocabularies efficiently.

6. POS (Part of Speech):
   Part of Speech (POS) refers to the grammatical category of a word based on its role and function in a sentence. POS tagging is an NLP process that assigns each word a grammatical label, helping machines understand sentence structure and meaning.

   NNS represents plural common nouns, meaning nouns that refer to more than one person, place, thing, or idea and are not proper names. These nouns typically take plural forms such as -s or -es.
   Example: students, cities, books

   NNP – Proper Noun, Singular
   NNP represents a singular proper noun, which is the name of a specific person, place, organization, or entity. Proper nouns are usually capitalized and refer to unique identifiers rather than general categories.
   Example: India, Google, Arnav

   VBD – Verb, Past Tense
   VBD represents a verb in the past tense, indicating an action or state that was completed in the past. These verbs often end with -ed, though many are irregular.
   Example: walked, studied, went

Universal POS Tags (UPOS – Standard NLP Tagset)

- ADJ – Adjective
- ADP – Adposition
- ADV – Adverb
- AUX – Auxiliary Verb
- CCONJ – Coordinating Conjunction
- DET – Determiner
- INTJ – Interjection
- NOUN – Noun
- NUM – Numeral
- PART – Particle
- PRON – Pronoun
- PROPN – Proper Noun
- PUNCT – Punctuation
- SCONJ – Subordinating Conjunction
- SYM – Symbol
- VERB – Verb
- X – Other / Unknown

Penn Treebank POS Tags (Complete & Detailed)

- CC – Coordinating Conjunction
- CD – Cardinal Number
- DT – Determiner
- EX – Existential *there*
- FW – Foreign Word
- IN – Preposition or Subordinating Conjunction

- JJ – Adjective
- JJR – Adjective, Comparative
- JJS – Adjective, Superlative
- LS – List Item Marker
- MD – Modal
- NN – Noun, Singular or Mass
- NNS – Noun, Plural
- NNP – Proper Noun, Singular
- NNPS – Proper Noun, Plural
- PDT – Predeterminer
- POS – Possessive Ending
- PRP – Personal Pronoun
- PRP$ – Possessive Pronoun
- RB – Adverb
- RBR – Adverb, Comparative
- RBS – Adverb, Superlative
- RP – Particle
- SYM – Symbol
- TO – *to*
- UH – Interjection
- VB – Verb, Base Form
- VBD – Verb, Past Tense
- VBG – Verb, Gerund or Present Participle
- VBN – Verb, Past Participle
- VBP – Verb, Non-3rd Person Singular Present
- VBZ – Verb, 3rd Person Singular Present
- WDT – Wh-Determiner
- WP – Wh-Pronoun
- WP$ – Possessive Wh-Pronoun
- WRB – Wh-Adverb

7. NER (Named Entity Recognition):
Named Entity Recognition is a Natural Language Processing (NLP) technique used to identify, classify, and label named entities in text into predefined categories such as person, organization, location, date, time, monetary value, and percentage.

NER helps machines understand who did what, where, and when, and is a core component of information extraction, question answering, search systems, and text analytics.