2. Byte-Pair Encoding (BPE) Tokenizer

    1. Understanding Unicode

        a. `chr(0)` returns ’`b'\x00`”

        b. `__repr__()` shows an escaped, unambiguous form whereas its printed representation outputs the character itself which is actual NULL character(invisible).

        c. When we add it to text, it simply adds an invisible character at that position which may not be visually noticeable.

    2. Unicode Encodings

        a. UTF-8 is preferred over UTF-16 or UTF-32 due to its superior compression, universal applicability without out of vocabulary errors, and native compability with byte-stream data.

        b. If we give the input 牛, it breaks and gives an error ”UnicodeDecodeError: ’utf-8’ codec can’t decode byte 0xe7 in position 0: unexpected end of data”. This is due to the fact that the decoding is done byte by byte and when the encoded character is multi-byte, it fails to decode it properly.

        c. `b'\xC3\x28`’ is invalid in UTF-8 because 0xC3 indicates the start of a 2-byte character, but 0x28 is not a valid continuation byte.

    3. BPE Training on TinyStories Training on the given tinystories dataset produces the following output:

```
(venv) sivasatvik@10-17-88-31 nyu-llm-reasoners-a1 %
time python student/train_bpe_tinystories.py
--input ./data/TinyStoriesV2-GPT4-train.txt
Elapsed (s): 61.72
RSS (MB): 158.3
Longest token length (bytes): 15
Longest token (latin-1):  accomplishment
python student/train_bpe_tinystories.py --input
319.31s user
14.65s system
539% cpu
1:01.86 total
```

    It seems that the longest token is ‘accomplishment’ with a length of 15 bytes. The memory usage is around 158.3 MB and the training took approximately 61.72 seconds. It makes sense that longer tokens are formed from frequently occurring sequences of characters in the dataset, and ‘accomplishment’ might be a common word in the TinyStories dataset.

3. Transformer Language Model Architecture

    1. Subsection 1

        a. a

        b. b

    2. Subsection 2

        a. a

b. b

c. c