Name: ____________________ NetID: ____________________

Worksheet 1: Tokenization

Tokenize and encode the following `sentence` as though you were a BERT-like language model. Use the `vocabulary` provided in the table below to convert the sentence into a sequence of `ids`. You may not necessarily use every token in the `vocabulary`.

1. Sequences begin with a [CLS] token and end with a [SEP] token (before padding).
2. Pad your sequence to 25 tokens in length.
3. All tokens in the `token` column have implicit leading whitespace *unless* they begin with “##”.

sentence: “how much wood could a woodchuck chuck if a woodchuck could chuck wood?”

vocabulary:

id	token
0	[MASK]
1	[PAD]
2	[CLS]
3	[SEP]
4	a
5	##chu
6	chuck
7	##ck
8	could
9	how
10	if
11	much
12	wood
13	?

Here’s an AI-generated image of a “woodchuck”, “groundhog”, or, officially, Marmota monax. *According to Wikipedia, the name “woodchuck” comes from the Algonquian or Narragansett word “wejack” or “wuchak”. They are relatively large rodents that burrow underground and hibernate through the winter. You may have seen them around on campus. Despite their superficial resemblance to a beaver, they do not “chuck wood.”*

ids:

[2, 9, 11, 12, 8, 4, 12, 5, 7, 6, 10, 4, 12, 5, 7, 8, 6, 12, 13, 3, 1, 1, 1, 1, 1]

Explanation:

Begin your sequence with a [CLS] token (2).

Every word in the sequence except “woodchuck” corresponds to a single token in the vocabulary. For “woodchuck”, you can’t encode it as [“wood”, “chuck”], since “wood” (12) and “chuck” (6) have implicit leading whitespace. Instead, encode “woodchuck” as [“wood”, “##chu”, “##ck”] (12, 5, 7).

End the sequence with a [SEP] token (3), then pad with five [PAD] tokens (1).

That’s it! (This is actually how BERT encodes the sentence, though with different numbers for the ids).