There are roughly 30k entries in the standard BERT vocabulary Download BERT vocabulary. In this HW, you should explore BERT's vocabulary to get a feel for what's really in there. Structure your exploration around the notion of how many "words" are really in its vocabulary. Follow along the path that's started in the notebook presented in lecture. Specifically, assume that the special reserved entries ([...]), numbers, subwords, and single characters are not words. From there you should explore the remaining full "words" to come up with a coherent rationale for your final count.

Submit a short report that describes your process for coming up with your estimate. You don't need to use unix commands to explore the vocabulary. Feel free to use Python, some other language, or an editor that supports regular expressions. If you make use of a word list or spelling dictionary please describe it.

```
# Import BERT
from transformers import BertTokenizer, BertModel
# Load the tokenizer by specifying the config for a model on the huggingface hub
tokenizer = BertTokenizer.from_pretrained('bert-base-uncased')
```

```
# Dump out the vocab, shows all entries
tokenizer.vocab.keys()
```

```
odict_keys(['[PAD]', '[unused0]', '[unused1]', '[unused2]', '[unused3]',
'[unused4]', '[unused5]', '[unused6]', '[unused7]', '[unused8]', '[unused9]',
'[unused10]', '[unused11]', '[unused12]', '[unused13]', '[unused14]',
'[unused15]', '[unused16]', '[unused17]', '[unused18]', '[unused19]',
'[unused20]', '[unused21]', '[unused22]', '[unused23]', '[unused24]',
'[unused25]', '[unused26]', '[unused27]', '[unused28]', '[unused29]',
'[unused30]', '[unused31]', '[unused32]', '[unused33]', '[unused34]',
'[unused35]', '[unused36]', '[unused37]', '[unused38]', '[unused39]',
'[unused40]', '[unused41]', '[unused42]', '[unused43]', '[unused44]',
'[unused45]', '[unused46]', '[unused47]', '[unused48]', '[unused49]',
'[unused50]', '[unused51]', '[unused52]', '[unused53]', '[unused54]',
'[unused55]', '[unused56]', '[unused57]', '[unused58]', '[unused59]',
'[unused60]', '[unused61]', '[unused62]', '[unused63]', '[unused64]',
'[unused65]', '[unused66]', '[unused67]', '[unused68]', '[unused69]',
'[unused70]', '[unused71]', '[unused72]', '[unused73]', '[unused74]',
'[unused75]', '[unused76]', '[unused77]', '[unused78]', '[unused79]',
'[unused80]', '[unused81]', '[unused82]', '[unused83]', '[unused84]',
'[unused85]', '[unused86]', '[unused87]', '[unused88]', '[unused89]',
'[unused90]', '[unused91]', '[unused92]', '[unused93]', '[unused94]',
'[unused95]', '[unused96]', '[unused97]', '[unused98]', '[UNK]', '[CLS]',
'[SEP]', '[MASK]', '[unused99]', '[unused100]', '[unused101]', '[unused102]',
'[unused103]', '[unused104]', '[unused105]', '[unused106]', '[unused107]',
'[unused108]', '[unused109]', '[unused110]', '[unused111]', '[unused112]',
'[unused113]', '[unused114]', '[unused115]', '[unused116]', '[unused117]',
'[unused118]', '[unused119]', '[unused120]', '[unused121]', '[unused122]',
'[unused123]', '[unused124]', '[unused125]', '[unused126]', '[unused127]',
'[unused128]', '[unused129]', '[unused130]', '[unused131]', '[unused132]',
'[unused133]', '[unused134]', '[unused135]', '[unused136]', '[unused137]',
```

```
'[unused138]', '[unused139]', '[unused140]', '[unused141]', '[unused142]',
'[unused143]', '[unused144]', '[unused145]', '[unused146]', '[unused147]',
'[unused148]', '[unused149]', '[unused150]', '[unused151]', '[unused152]',
'[unused153]', '[unused154]', '[unused155]', '[unused156]', '[unused157]',
'[unused158]', '[unused159]', '[unused160]', '[unused161]', '[unused162]',
'[unused163]', '[unused164]', '[unused165]', '[unused166]', '[unused167]',
'[unused168]', '[unused169]', '[unused170]', '[unused171]', '[unused172]',
'[unused173]', '[unused174]', '[unused175]', '[unused176]', '[unused177]',
'[unused178]', '[unused179]', '[unused180]', '[unused181]', '[unused182]',
'[unused183]', '[unused184]', '[unused185]', '[unused186]', '[unused187]',
'[unused188]', '[unused189]', '[unused190]', '[unused191]', '[unused192]',
'[unused193]', '[unused194]', '[unused195]', '[unused196]', '[unused197]',
'[unused198]', '[unused199]', '[unused200]', '[unused201]', '[unused202]',
'[unused203]', '[unused204]', '[unused205]', '[unused206]', '[unused207]',
'[unused208]', '[unused209]', '[unused210]', '[unused211]', '[unused212]',
'[unused213]', '[unused214]', '[unused215]', '[unused216]', '[unused217]',
'[unused218]', '[unused219]', '[unused220]', '[unused221]', '[unused222]',
'[unused223]', '[unused224]', '[unused225]', '[unused226]', '[unused227]',
'[unused228]', '[unused229]', '[unused230]', '[unused231]', '[unused232]',
'[unused233]', '[unused234]', '[unused235]', '[unused236]', '[unused237]',
'[unused238]', '[unused239]', '[unused240]', '[unused241]', '[unused242]',
'[unused243]', '[unused244]', '[unused245]', '[unused246]', '[unused247]',
'[unused248]', '[unused249]', '[unused250]', '[unused251]', '[unused252]',
'[unused253]', '[unused254]', '[unused255]', '[unused256]', '[unused257]',
'[unused258]', '[unused259]', '[unused260]', '[unused261]', '[unused262]',
'[unused263]', '[unused264]', '[unused265]', '[unused266]', '[unused267]',
'[unused268]', '[unused269]', '[unused270]', '[unused271]', '[unused272]',
'[unused273]', '[unused274]', '[unused275]', '[unused276]', '[unused277]',
'[unused278]', '[unused279]', '[unused280]', '[unused281]', '[unused282]',
'[unused283]', '[unused284]', '[unused285]', '[unused286]', '[unused287]',
```

The first 999 tokens appear to be reserved, and most are of the form [unusedXXX]. We have 999 special reserved entries

```
vocab_size = len(tokenizer.vocab.keys())
vocab_size
```

```
30522
```

```
# Filter out single characters
one_chars = []

# For each token in the vocabulary...
for token in tokenizer.vocab.keys():

    # Record any single-character tokens.
    if len(token) == 1:
        one_chars.append(token)

print('Number of single character tokens:', len(one_chars), '\n')
```

```
Number of single character tokens: 997
```

We have 997 single character tokens

```
# Filter out subwords
num_subwords = 0

# For each token in the vocabulary...
for token in tokenizer.vocab.keys():

    # If it's a subword...
    if len(token) >= 2 and token[0:1] == '#':

        # Tally all subwords
        num_subwords += 1

print('Number of subwords: {:,}'.format(num_subwords))
```

```
Number of subwords: 5,828
```

We have 5828 subwords

```
# Filter out numbers
count = 0

# For each token in the vocabulary...
for token in tokenizer.vocab.keys():

    # Tally if it's a number.
    if token.isdigit():
        count += 1

print('Vocab includes {:,} numbers.'.format(count))
```

```
Vocab includes 881 numbers.
```

We have 881 numbers

In conclusion:

- The original bert base uncased vocal has 30522 entries:
    - it has 999 special reserved entries
    - it has 881 numbers
    - it has 5828 subwords
    - it has 997 single characters

Thus, the remaining full "words" count will be 21817.