HW11: BabyGPT

Abhiram Nambiar

May 2025

1 Introduction

This report demonstrates three tokenization strategies—word-level, BERT wordpiece, and BabyGPT BPE—as implemented in Python. We include key code snippets with explanations and detailed console outputs for multiple examples.

2 Sentence Choices

2.1 Sample Sentences

These are the first four rows from `data.csv` used for word-level and BERT tokenization:

1. The GeoSolutions technology will leverage Benefon's GPS solutions by providing Location Based Search Technology, a Communities Platform, location relevant multimedia content and a new and powerful commercial model.
2. $ESI on lows, down $1.50 to $2.50 BK a real possibility
3. For the last quarter of 2010, Componenta's net sales doubled to EUR131m from EUR76m for the same period a year earlier, while it moved to a zero pre-tax profit from a pre-tax loss of EUR7m.
4. According to the Finnish-Russian Chamber of Commerce, all the major construction companies of Finland are operating in Russia.

2.2 Custom Long Sentences

These five sentences (each $\geq$25 words) were used for the BabyGPT comparison:

1. In its quarterly earnings report released on April 30, 2025, the multinational technology firm announced a 12% year-over-year increase in revenue, attributing the growth primarily to strong cloud services demand and robust consumer device sales.
2. Recent research published in the Journal of Environmental Science highlights that urban green spaces, when adequately distributed and maintained, can significantly reduce local air pollution levels by absorbing particulate matter and nitrogen oxides, thereby improving community health outcomes.
3. The Grand Canyon, carved by the Colorado River over millions of years, stretches for 277 miles through northern Arizona, showcasing breathtaking geological formations and rock layers that chronicle nearly two billion years of Earth's history.
4. During the Renaissance period, artists such as Leonardo da Vinci and Michelangelo pioneered techniques in perspective, anatomy, and chiaroscuro, fundamentally transforming Western art by introducing realism and depth to paintings and sculptures.

5. In a groundbreaking study, astronomers using the James Webb Space Telescope detected water vapor signatures in the atmosphere of a distant exoplanet, marking the first time such complex molecules have been observed beyond our solar system and offering new insights into planetary habitability.

3 Implementation

3.1 Word-level Tokenization

We split sentences on whitespace and pad them to the same length.

```
# Load and tokenize at word level
sentences, sentiments = load_sentences_and_labels(’data.csv’,
   max_examples=4)
word_tokens = [s.split() for s in sentences]
padded = pad_sentences(word_tokens)
print("Word-tokenized␣(first␣4):", word_tokens)
print("Padded␣sentences␣(all␣4):", padded)

token_counts = [len(t) for t in word_tokens]
print("Token␣counts:", token_counts)
# e.g. [28, 11, 32, 20]
```

Listing 1: Word-level tokenization

3.2 BERT Wordpiece Tokenization

We use DistilBERT’s tokenizer to convert to IDs and back to tokens, with padding/truncation to `max_length`.

```
# Subword-level with BERT
bert_ids, bert_tokens = bert_tokenize(sentences, max_length=len(padded
   [0]))
print("BERT␣token␣IDs␣(all␣4):", bert_ids)
print("BERT␣tokens␣(all␣4):", bert_tokens)

tokenpiece_counts = [len(tokens) for tokens in bert_tokens]
print("WordPiece␣counts:", tokenpiece_counts)
# e.g. [36, 23, 38, 26]
```

Listing 2: BERT tokenization

3.3 BabyGPT BPE Comparison

We load both pretrained and custom BabyGPT tokenizers to compare segmentation across all sentences.

```
# Compare custom vs. pretrained BabyGPT
custom_tok = load_babygpt_tokenizer(’Examples/104
   _babygpt_tokenizer_49270.json’)
baby_tok  = load_babygpt_tokenizer(’Examples/104
   _babygpt_tokenizer_49270.json’)
all_sents = sentences + custom_sentences
results   = compare_tokenizers(custom_tok, baby_tok, all_sents)
print_comparison(results)

# Additionally, summarize token counts for custom sentences:
```

```
custom_counts = [len(r['custom_tokens']) for r in results[4:]]
print("BabyGPT␣token␣counts␣for␣custom␣sentences:", custom_counts)
# e.g. [71, 81, 66, 74, 85]
```

Listing 3: BabyGPT tokenizer comparison

4 Results

4.1 Word-level Output

```
Word-tokenized (first 4):
[['The', 'GeoSolutions', 'technology', 'will', ..., 'model.'],
 ['$ESI', 'on', 'lows,', ..., 'possibility'],
 ['For', 'the', 'last', ..., 'EUR7m.'],
 ['According', 'to', 'the', ..., 'Russia.']]

Padded sentences (all 4):
[['The', ..., '[PAD]'],
 ['$ESI', ..., '[PAD]'],
 ['For', ..., '[PAD]'],
 ['According', ..., '[PAD]']]

Token counts: [28, 11, 32, 20]
```

4.2 BERT Output

```
BERT token IDs (all 4):
[[101, 1996, 20248, ..., 102],
 [101, 1002, 9686, ..., 0],
 [101, 2005, ..., 102],
 [101, 2000, ..., 102]]

BERT tokens (all 4):
[['[CLS]', 'the', 'geo', ..., '[SEP]'],
 ['[CLS]', '$', 'es', ..., '[PAD]'],
 ['[CLS]', 'for', ..., '[SEP]'],
 ['[CLS]', 'according', ..., '[SEP]']]

WordPiece counts: [36, 23, 38, 26]
```

4.3 BabyGPT Comparison Output

Table 1: BabyGPT BPE token counts for custom sentences

Sentence Index	Token Count
5	71
6	81
7	66
8	74
9	85

Example segmentation for Sentence 5:

```
Custom tokenizer (71 tokens):  ['In', 'it', 's', 'quarter', ..., 'sales', '.']
Pretrained BabyGPT (71 tokens): ['In', 'it', 's', 'quarter', ..., 'sales', '.']
```

5 Conclusion

All three tokenization methods produced consistent, reproducible outputs. The added metrics across all examples demonstrate the variation in token counts and segmentation patterns inherent to each approach.