唐一早
2100012613

Homework 2

自然语言处理基础
2024 年 4 月 28 日

Lab2 Report

2024 年 4 月 28 日

1 Problem 1

1.1 Q1

最中得到的词汇表大小为 16050，原训练集的单词数量为 107259，经过 BPE 方法 tokenize 后得到的 token 数量为 124695。

1.2 Q2

通过 BPE 方法的得到的测试数据的总 token 数量为 2092，其中 <unk> 字符数量为 105，且均非英文字符。事实上因为 BPE 方法是从字符开始不断结合 bigram 来进行 tokenization。因此只要是用同一种字符构筑的语言都可以被识别并且被 tokenize，只有用 tokenizer 没见过的字符才会被识别成 <unk>。而普通的按照空格或符号分词的方法一旦遇到未出现过的单词就会被分为 <unk>，更不用说未见过的字符和语言。所以 BPE 方法得到 <unk> 的概率一定比按单词分词的方法好。

2 Problem 2

2.1 Method

2.1.1 BPE

如 problem1 中所实现的，从字母开始，不断找词频最高、且连续的两个 token 合并，直到达到目标词数。没有空格的语言也可以分词，没遇见过的 word 会按出现频率最高的 subword，但是遇到没有遇见过的语言会全部识别为 unk。

2.1.2 BBPE

BBPE 核心思想将 BPE 的从字符级别扩展到子节（Byte）级别。采用 BBPE 的好处是可以跨语言共用词表，显著压缩词表的大小。而坏处就是，对于类似中文这样的语言，一段文字的序列长度会显著增长。因此，BBPE 可能比 BPE 表现的更好。然而，BBPE sequence 比起 BPE 来说略长，这也导致了更长的训练/推理时间。

2.1.3 WordPiece

WordPiece 算法可以看作是 BPE 的变种，采用的不是 bigram 的频率而是两个 subtoken 之间的互信息作为标准。因此其分词结果会和 BPE 类似。

2.1.4 Unigram

Unigram Language Model 先初始一个大词表，接着通过语言模型评估 subword 概率不断减少词表，直到限定词汇量。显然因为最终没有显示的 tokenize 方法，要对单词尝试所有种类的 tokenize 可能，这种方法在 tokenize 和训练速度上比较慢。

2.1.5 SentencePiece

SentencePiece 是把一个句子看作一个整体，再拆成片段，而没有保留天然的词语的概念。一般地，它把空格也当作一种特殊字符来处理，再用 BPE 或者 Unigram 算法来构造词汇表。因而有没有空格对 SentencePiece 来说没什么分别，所以也可以做中文分词。根据使用算法不同，在 unk 字符数量和数字 tokenize 上表现不同。

Model	Type of Tokenizer
fast MPNet	WordPiece
PhoBERT	Byte-Pair-Encoding
T5	SentencePiece
fast T5	Unigram
fast MBART	BPE
fast PEGASUS	Unigram
PEGASUS	SentencePiece
XLM	Byte-Pair-Encoding
TAPAS	WordPiece
BertGeneration	SentencePiece
BERT	WordPiece
fast BERT	WordPiece
XLNet	SentencePiece
GPT-2	byte-level Byte-Pair-Encoding
fast XLNet	Unigram
fast GPT-2	byte-level Byte-Pair-Encoding
fast ALBERT	Unigram
ALBERT	SentencePiece
CTRL	Byte-Pair-Encoding
fast GPT	Byte-Pair-Encoding
Flaubert	Byte-Pair Encoding
FAIRSEQ	Byte-Pair Encoding
Reformer	SentencePiece
fast Reformer	Unigram
Marian	SentencePiece

图 1: 各个语言模型所使用的 tokenization 方法

2.2 Examples

所有案例在 test.ipynb 中测试, 具体案例可以前往查看

2.2.1 Bert

Bert 使用 wordpiece 方法，在分词时会按标点和空格先分词，再对单词分词。因而“don't”会被分为“don”“'”“t”。在中文分词上是一个字一个字分，在数字上会把小数拆开，会把较长的数字分为较短的数字。

2.2.2 gpt

gpt 系列使用的是 BBPE 方法，在分词时也会按标点和空格先分词，再对单词分词。中文和数字上分词效果和 bert 类似，区别是 gpt1 对长数字分词每个小数字的长度比 gpt2 短一些

2.2.3 t5 & XLNet

t5 和 XLNet 采用的是 SentencePiece 方法，分词结果类似，在英文分词里偶尔会拆出一些单个字母，分数字情况和以上类似，分中文时会将标点之间的所有中文分为一个词。

2.2.4 XLM & Qwen & Llama

这三个模型采用的都是 BPE 方法，英文分词结果与 BBPE 类似，在中文分词时，都可以将一些类如“尽管”这样的常用词结合为一个 token，但是 qwen 在对数字分词时会将数字拆分为一个一个数字，其他与上面类似。

2.2.5 mBart

mBart 模型使用的说 unigram 方法，其结果与 XLM 类似，也可以中文分词时，将一些常用词结合为一个 token。值得注意的是因为 unigram 分词较慢，同样长度的句子，mBart 分词时间明显较长。