BPE を用いたトークナイザーの性能に対する, 言語・語彙数・データセットの影響

中島大　野崎雄太　佐藤諒　麻場直喜　川村晋太郎
株式会社リコー　デジタル戦略部
デジタル技術開発センター　言語 AI 開発室
{dai.nakashima, yuta.nozaki1, ryo.sato4, naoki.asaba, shintaro.kawamura}@jp.ricoh.com

概要

大規模言語モデル (Large Language Model: LLM) への入出力に使われているトークナイザーについて, 下流タスクから評価した研究が複数行われているものの, タスク内容やモデルの種類など議論の難しい要素が非常に多く, 実際の開発では経験的な決め打ちによってトークナイザーを作成している場合が多いと推察する. そこで, 本研究ではトークナイザーの定量的な指標となる, 評価用テキストをトークナイズしたときの 1 トークンあたりの平均文字数 (Length per Token: LPT) を複数の場合で調べた. 結果として, LPT は語彙数に対していずれの言語でも log 的に増加し, ただしその係数は言語特性を強く反映することが明らかとなった. また, 英語に比べて日本語の場合で特に LPT がデータセットのドメインに影響を受けることが明らかとなった.

1　はじめに

近年, LLM の研究・開発が盛んに行われている. その中でトークナイザーは LLM にテキストを入出力するための不可欠な要素である. 一方で, 複数の研究結果 [1, 2] から, トークナイザーの語彙数は下流タスクに対して強く良く影響するものではないと判断できる. そのため, トークナイザーの語彙数を大幅に増やした LLM は主流ではなく, 現在は語彙数 32,000 程度のトークナイザーが広く用いられている. これは 16-bit に収まる値でもある ($< 65,535$). 例外として多言語モデルではより多くの語彙数が必要とされる. 実際に多言語モデルの Qwen[3] では語彙数 150,000 となっており, 最も小さい 7B モデルでは語彙数の影響を受けているパラメータが全体の 8.9% を占めている. また, LLM の学習に用いるテキストはトークナイザーによってトークン数が変わる. 以上のようにしてトークナイザー, 特にその語彙数は LLM の学習コストに影響を与える.

本研究ではそうした観点から, トークナイザーの性能評価として, 評価用テキストをトークナイズしたときの 1 トークンあたりの文字数 (LPT) を複数の場合で調べた. その結果について, 特に以下の 2 点に注目して議論する.

- 語彙数・言語による影響
- データセットによる影響

1.1　Length per Token (LPT)

LPT は Characters per Token とも呼ばれ, トークナイザーの性能を定量的に評価する指標である. 与えられたテキストをトークナイズしたときに 1 トークンあたり何文字であるかを表しており, テキストの文字数をトークン長で割った値である. これはテキストの圧縮比の逆数に相当する. これが大きいほどトークン化効率が良いと言え, LPT が高いほど固定されたトークン数で扱える文字数が増える, とも言い換えられる. ここで, LPT は学習に用いていない評価用のデータセットをトークナイズしたときの 1 トークンあたりの平均文字数であり, トークナイザーの語彙辞書内の語彙の平均文字数ではない. LPT は下流タスクの性能を表すものではないが, コストの面で LLM と直接関係している.

本研究ではトークナイザーの手法として Byte Pair Encoding (BPE) を採用している [4]. 他の類似手法として Unigram[5] も BPE と同等かそれ以上の性能を持つことが知られているが [6], 本研究では Llama[7] と同じ BPE を SentencePiece[8] で実装した.

2　語彙数・言語による影響

LPT が語彙数と正の相関を持つことは明らかだが, 語彙数を 2 倍にしても LPT が 2 倍になるわけで

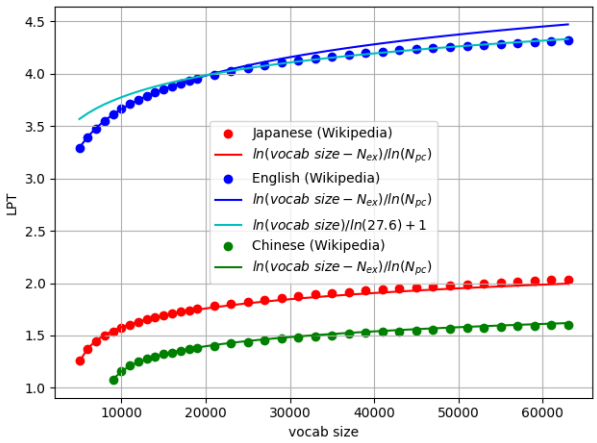

図 1　言語ごとの語彙数と LPT. 青点線が英語, 赤点線が日本語, 緑点線が中国語の結果を表し, 各色の実線がその fit 結果を表す. fit は $vocab\ size < 20000$ で行い, そのパラメータは表 1 に示す.

表 1　フィットしたパラメータの値

	N_{pc}	N_{ex}
English	11.7728	1634.19
Japanese	245.88	4032.83
Chinese	853.231	7520.37

はない. また, LPT は言語によってその値が非常に大きく変化する. 例えば英語での典型的な LPT は 4 程度だが, 日本語では 2 程度である. ここでは, 語彙数と言語が, トークナイザーの性能にどのように影響するかを示す.

2.1　実験方法

日本語・中国語・英語について, Wikipedia のコーパスを用いてトークナイザーを作成した. このとき, 日本語・中国語についてはデータ全体, 英語については全体から 3 GB 相当の行をランダムにサンプルし, そこから 98% を学習用, 2% を評価用として BPE の学習・評価を行った.

2.2　実験結果

図 1 は英語・日本語・中国語での語彙数 ($vocab\ size$) と LPT の関係を表しており, 実線はそれぞれの言語で, 5,000, 5,000, 9,000 から 20,000 までの範囲のみでフィットさせた. 図 1 より, いずれの言語でも語彙数に対して同様に LPT が log 的に増加していくことが分かり,

$$(vocab\ size) = \left(N_{pc}\right)^{(LPT)} + N_{ex} \tag{1}$$

で非常によく fit できている. これは N_{pc} の文字の組み合わせでほとんど全ての語彙が形成され, N_{ex} が語彙に寄与しない低頻度の文字であるとしたときの LPT と vocab size の関係式である. この関係式は, $character\ coverage \sim 1$ のとき $LPT = 1$ となる語彙数はコーパス内の文字の種類数とおよそ同じになることから

$$vocab\ size\ (\mathrm{LPT} = 1) \sim \text{コーパス内の文字種} \tag{2}$$

のように $LPT = 1$ のときの語彙数が固定されていることと, 関係が log 的であることから決定した. なお, 英語については日本語・中国語と特性が異なり, 語彙数の大きな領域でやや異なる振舞いにみえる. この領域では一例として

$$(vocab\ size) = 27.6^{(LPT)-1} \tag{3}$$

でよく fit されているように見えるものの, さらに大きな語彙数の領域を調査する必要があり, 今回は行わない.

以上の結果から, 日本語・中国語は N_{pc} が英語と比べて大きいために, 語彙数を増やしても LPT が上昇しにくいことが分かる. また, LPT を 2 倍するには語彙数を 2 乗する必要がある ($vocab\ size \gg N_{ex}$ の場合).

最後に fit の妥当性について考察する. N_{ex} については (2) 式の関係があるため容易で, 英語の $N_{pc} + N_{ex}$ に日本語のひらがな 84 字カタカナ 84 字, 常用漢字数 2,136 字を加えると 3,770 であり, 日本語の $N_{pc} + N_{ex}$ とおよそ一致する. 同様に英語の N_{ex} に, 中華人民共和国国務院の通用規範漢字表 (Retrieved 2023-01-09) に記載されている文字数 8,105 を加えると 9,739 となり, ややオーバーするが中国語の $N_{pc} + N_{ex}$ とおよそ一致する.

N_{pc} については, 英語の N-gram (bigram, trigram 等) が古くから研究されており [9, 10], 英語語彙を形成する文字には偏りがあることが知られている. 例として, 以下に頻出の bigram, trigram を示す.

表 2　英語の bigram, trigram[10]

N-grams	Examples
1	e, t, a, o, i, n, s, r, h, l, ···
2grams	th, he, in, er, an, re, on, at, en, nd, ···
3grams	the, and, ing, ion, tio, ent, ati, for, her, ter, ···

このような偏りがトークナイザーにも反映されており, トークナイザーの語彙にも同様の偏りが見られる (図 2).

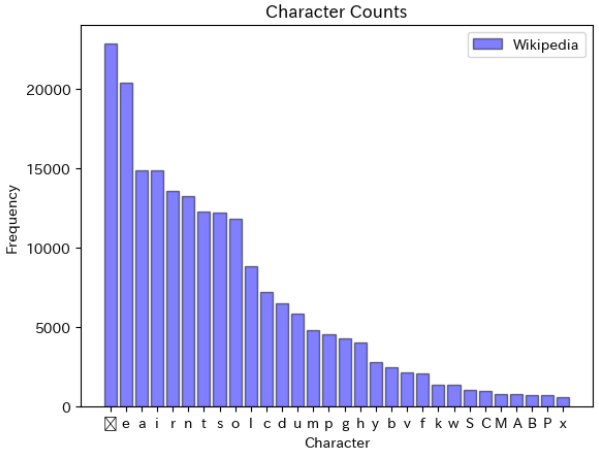

図 2　トークナイザーの語彙における頻出文字と, その頻度 (上位 30 種類のみ.). "▁"は, 空白文字を表す. なお, この図はコーパス内の文字の出現頻度ではなく, トークナイザーの語彙辞書内での出現頻度である.

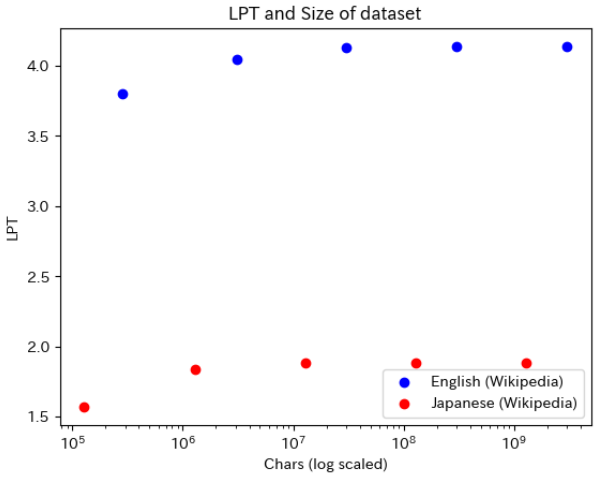

図 3　データセットの文字数と, LPT の相関. 英語の文字数はそのままファイルサイズに相当する.

3　データセットによる影響

ここでトークナイザーの学習に使うデータセットについて検証する. 検証すべき内容は, トークナイザーの学習にどの程度の量のデータセットが必要で, またデータセットにはどのようなものを用意すれば良いか, という 2 点である.

3.1　実験方法

データセットの量について

英語, 日本語の Wikipedia からランダムに行をサンプルし, 語彙数を 32,000 として BPE で学習した.

データセットの内容について

英語・日本語の公開されているデータセットとして, 表 3 の学習データセット・評価データセットを用意し, それぞれの学習データセットに対して個別に 32,000 語彙のトークナイザーを作成, 評価データセットで評価した [11, 12, 13, 14, 15, 16]. 次節に記すデータセットの量についての実験から 300 MB のデータがあれば十分と判断し, これを用意した. ここで日本語は基本的に 1 文字 3 バイトであるため同じファイルサイズでも文字数は少なくなるが, それでも 300 MB あれば十分であった.

表 3　データセット

英語	Wikipedia, Book (RedPajama), C4
日本語	Wikipedia, CC100, mC4, OSCAR

3.2　実験結果

3.2.1　データセットの量について

データセットの量 (文字数) と LPT の相関を図示したものが図 3 である. このように, ある程度まではデータセットを大きくするほど性能が伸びていくが, ある程度を超えると性能の伸びが急激に低下する. 図 3 において, およその閾値は文字数 10^6 であり, 英語では 1 MB に相当する. ただし, この結果はデータセットの質・ドメイン・言語差に影響を受ける.

3.2.2　データセットの内容について

トークナイザーの学習に用いるデータセットによって, その評価結果が変わることが表 4, 5 から分かる. 日本語 Wikipedia で評価する場合, 学習データセットを CC100 から Wikipedia にすることは, 語彙数を 14,000 から 20,000 に変更することに相当するスコアの向上が見られる. また, 特徴的な点として言語による差異が見られる. 表 4 は英語, 表 5 は日本語でそれぞれ学習・評価した結果である. 特にこの中で英語での Wikipedia と C4, 日本語での Wikipedia と mC4 に注目すると, 英語で最も相対スコアの低いパターンは, Wikipedia で学習し, C4 で評価したものである. 日本語で最も相対スコアの低いパターンも同様で, Wikipedia で学習し, mC4 で評価したものが 4 種の中では最も低い. しかしそのスコアは大きく異なり, 英語では相対スコアが 0.968 であるのに対して日本語では相対スコアが 0.874 まで低下しており, スコアにおよそ 1 割の差がある. 実験 2 の結果を思

い出せば, 英語では bigram, trigram 等の語彙を構成する主要な文字 (T, H, E, ・・・) がコーパスに依存しないのに対し, 日本語では語彙を構成する主要な文字がコーパスのドメインによって置き換わっているためと考えられる.

図4, 5は英語 Wikipedia と英語 C4, 日本語 Wikipedia と日本語 mC4 でそれぞれ作成したトークナイザー (32,000 語彙) の語彙辞書に出現する文字を表しており, コーパスの違いがトークナイザーにどのように影響するのかを示す. 図 4 から英語 Wikipedia と C4 ではそれぞれに由来するトークナイザーの語彙における各文字の出現頻度はほとんど変化がない一方で, 図 5 から日本語 Wikipedia と mC4 の場合では, トークナイザーの語彙に出現する文字の頻度に大きな変化があることが分かる. また, トークナイザーの語彙自体も英語では 32,000 語彙のうち 72% が共通であるのに対して日本語では 54% にとどまっていた. 以上の結果から, 英語のトークナイザーと比べて, 日本語のトークナイザーを作成する場合は, その言語特性のためにデータセットのドメインに気を付ける必要がある.

表 4 学習データセットと評価データセット (英語). 数値は LPT で, 括弧 () 内の数値は同種の評価データでの LPT の値を 1 としたときの相対スコア.

	学習データ		
	wikipedia	book	C4
wikipedia	4.13 (1)	4.02 (0.972)	4.05 (0.979)
book	3.83 (0.968)	3.95 (1)	3.88 (0.983)
c4	4.21 (0.968)	4.25 (0.978)	4.35 (1)

表 5 学習データセットと評価データセット (日本語). 数値は LPT で, 括弧 () 内の数値は同種の評価データでの LPT の値を 1 としたときの相対スコア.

	学習データ			
	wikipedia	CC100	mC4	OSCAR
wikipedia	1.88 (1)	1.70 (0.904)	1.70 (0.902)	1.73 (0.919)
cc100	1.93 (0.853)	2.26 (1)	2.16 (0.955)	2.21 (0.979)
mC4	1.71 (0.874)	1.83 (0.938)	1.95 (1)	1.90 (0.975)
OSCAR	1.80 (0.885)	1.97 (0.969)	2.00 (0.983)	2.03 (1)

4 おわりに

以上, トークナイザーの性能に対する言語・語彙数・データセットの影響の調査結果を示した. 結果として, 実験 2 において, 今回調査した英語・日本語・中国語のいずれでも LPT は語彙数に対して log で増加し, そのパラメータは言語の特性を反映していることが明らかとなった. また, 実験 3 において, データセットのサイズとトークナイズ性能の相関,

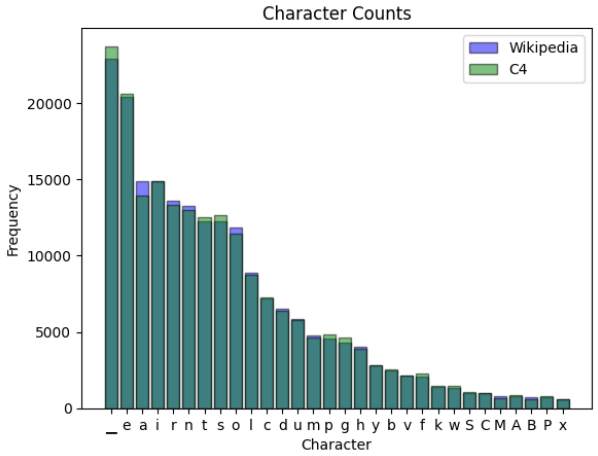

図 4 英語 Wikipedia, C4 でそれぞれ作成したトークナイザーの語彙における頻出文字と, その頻度 (Wikipedia における上位 30 種類のみ.).

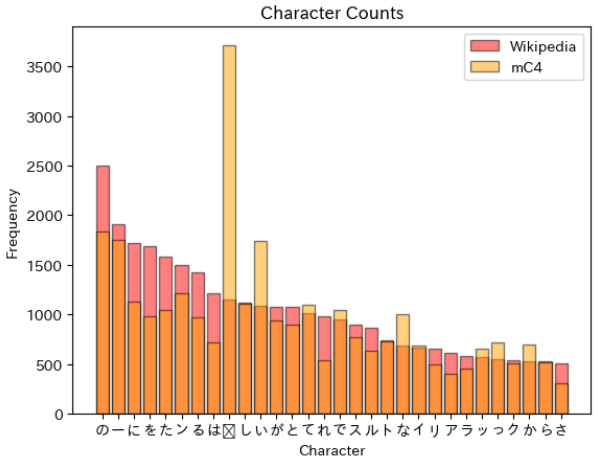

図 5 日本語 Wikipedia, mC4 でそれぞれ作成したトークナイザーの語彙における頻出文字と, その頻度 (Wikipedia における上位 30 種類のみ.). マス文字は空白文字を表す.

及びデータセットの内容による影響を調べた結果, 特に日本語のトークナイザーは英語と比べてデータセットの内容に強く影響を受けることが明らかとなった. 結論としてトークナイザーは言語特性に影響を受け, そのために日本語のトークナイザーを作成する場合はデータセットの内容の偏りに注意する必要がある.

本研究は Ricoh-13B 日英バイリンガルモデル [18] および, より大規模な 70B モデル作成のために行った.

参考文献

[1] 井上誠, Nguyen Tung, 中町礼文, 李聖哲, 佐藤敏紀, 日本語 GPT を用いたトークナイザの影響の調査, 言語処理学会第 28 回年次大会 (2022).

[2] Thamme Gowda, and Jonathan May Finding the Optimal Vocabulary Size for Neural Machine Translation, EMNLP 2020, 3955 (2020).

[3] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu, Qwen Technical Report. arXiv: 2309.16609 (2023).

[4] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural Machine Translation of Rare Words with Subword Units, In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1715–1725, Berlin, Germany. Association for Computational Linguistics (2016).

[5] Taku Kudo, Subword Regularization: Improving Neural Network Translation Models with Multiple Subword Candidates, In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 66–75, Melbourne, Australia. Association for Computational Linguistics (2018).

[6] 藤井巧朗, 柴田幸輝, 山口篤季, 十河泰弘, 日本語 Tokenizer の違いは下流タスク性能に影響を与えるか？, 言語処理学会第 29 回年次大会 (2023).

[7] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample, LLaMA: Open, Efficient Foundation Language Models, arXiv: 2302.13971 (2023).

[8] Taku Kudo and John Richardson, SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing, In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 66–71, Brussels, Belgium. Association for Computational Linguistics (2018).

[9] Henry Beker and Fred Piper, CIPHER SYSTEMS: The Protection of Communications (Northwood Books, London, 1982).

[10] Peter Norvig, English Letter Frequency Counts: Mayzner Revisited or ETAOIN SRHLDCU, norvig.com. Retrieved 2023-01-09.

[11] Common Crawl, https://commoncrawl.org.

[12] Together Computer, RedPajama: An Open Source Recipe to Reproduce LLaMA training dataset https://github.com/togethercomputer/RedPajama-Data (2023).

[13] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu, Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, Journal of Machine Learning Research, 21(140): 1–67 (2020).

[14] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov, Unsupervised Cross-lingual Representation Learning at Scale, In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 8440–8451, Online. Association for Computational Linguistics (2020).

[15] Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave, CCNet: Extracting High Quality Monolingual Datasets from Web Crawl Data, In Proceedings of the Twelfth Language Resources and Evaluation Conference, pages 4003–4012, Marseille, France. European Language Resources Association (2020).

[16] OSCAR, https://oscar-project.org.

[17] Bo Zheng, Li Dong, Shaohan Huang, Saksham Singhal, Wanxiang Che, Ting Liu, Xia Song, and Furu Wei. Allocating Large Vocabulary Capacity for Cross-Lingual Language Model Pre-Training, In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 3203–3215, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics (2021).

[18] 麻場直喜, 野崎雄太, 中島大, 佐藤諒, 池田純一, 伊藤真也, 近藤宏, 小川武士, 坂井昭一朗, 川村晋太郎, 語彙置換継続事前学習による日英バイリンガルモデルの構築と評価, 言語処理学会第 30 回年次大会 (2024, 予定).