

# 多言語事前学習モデルのための SentencePiece トークナイザーへのサブワード追加

今村賢治 隅田英一郎

国立研究開発法人 情報通信研究機構

{kenji.imamura,eiichiro.sumita}@nict.go.jp

## 概要

多言語事前学習モデルは、トークナイザーが確定した上で事前学習されているため、多言語事前学習モデルを新規言語に拡張する際は、トークナイザーも同時に修正する必要がある。本稿では、多言語事前学習モデルを新規言語（本稿ではイヌイット語）に対応させるため、SentencePiece トークナイザーに新規サブワードを追加する。我々の実験では、既存言語のサブワード分割を変更せずに、イヌイット語のサブワード分割が可能となり、mBART-50 事前学習モデルを英語・イヌイット語翻訳に適用することができた。

## 1 はじめに

近年、さまざまな事前学習モデルがリリースされている。その中でも、mBERT [1], XLM-RoBERTa [2], mBART [3, 4], mT5 [5] などの多言語モデルは、複数の言語を一つのモデルで学習されているため、言語横断分類や機械翻訳に有効である。しかし、これらの事前学習モデルは、語彙が事前学習時に決定されている。事前学習モデル自体の語彙は、追加するだけなら、単語埋込テーブルを拡張することで可能である [6]。しかし、一般的には、事前学習モデルはトークナイザー（とそのモデル）を確定した上で事前学習されているため、事前学習モデルの語彙を変更する場合、同時にトークナイザーも修正する必要がある。

たとえ新規言語であっても、トークナイザーがサポートしている文字を使用する言語の場合は、（たとえ最適でないとしても）既存モデルが流用可能である。一方、トークナイザーがサポートしていない文字が使われている言語の場合、未知語となってしまいうため、何らかの対処が必要となる。

本稿では、多言語事前学習モデルのトークナイ

ザーに焦点を当てる。具体的には、mBART-50 [7] を新規言語に対応させるため、SentencePiece トークナイザー [8, 9] に新規サブワードを追加する。タスクは、国際会議 WMT-20 [10] のニュース翻訳タスクで対象となったイヌイット語・英語翻訳である。mBART-50 も、それに付随する SentencePiece も、モデルにはイヌイット語を含んでいない。

われわれの目標は、新規言語に事前学習モデルを適用するために、既存言語のトークナイズ結果を極力変えないようにしながら、新規言語に対してサブワード化方法を提供することである。既存言語の解析結果を変えないことによって、事前学習モデルの効果を最大限活用できる。

## 2 SentencePiece トークナイザー

SentencePiece は入力文字列をそのままサブワード分割できるトークナイザーである（無損失符号化と呼ばれている）。英語のように単語区切りがスペース文字によって明示されている言語と、日本語、中国語のように単語間に区切り文字がない言語を混在して扱うことができるため、多言語事前学習モデルのトークナイザーとして採用されている。

SentencePiece は、バイトペア符号化 [11] などのサブワード方式もサポートしているが、本稿では、mBART-50 で使用され、SentencePiece のデフォルトとなっているユニグラムモデルについて考える。

### 2.1 解析方式/サブワード符号化

SentencePiece では、無損失符号化を実現するため、前処理でスペースを特殊文字（デフォルトはユニコードの U+2581）に置換して単語区切りがない文字列にしたのちに、以下の方法でサブワード列に分割する（図 1） [12]。ちなみにこの方法は、日本語の形態素解析方法（たとえば MeCab [13]）と同じである。

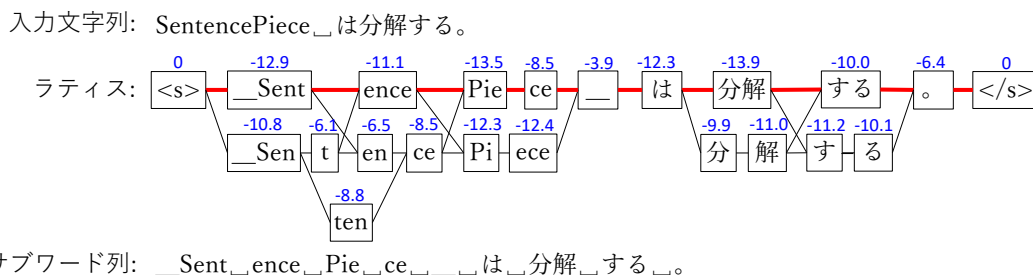


図1 SentencePiece ユニグラムモデルによるサブワード符号化の例。' ' はスペース文字 (U+0020) を表す。ラティスの各サブワードの上の値はユニグラム対数尤度、赤線はビタビパスを表す。

1. 入力テキストとユニグラムモデル（形態素解析における辞書相当）のサブワードを照合し、すべてのサブワード候補を取得、ラティス構造に配置する。
2. ラティスに対してビタビ探索を行い、最も尤度の高いパスを探索する。最尤パス上のサブワード列を出力する。

## 2.2 ユニグラムモデルの学習

ユニグラムモデルの学習は、隠れマルコフモデル (HMM) の教師なし学習 [12] を応用したものである。以下の手順で行う。基本的に、初期モデルのサブワードを削除することで所定の語彙サイズを得る方式であるので、最終モデルの語彙は、初期モデルの語彙のサブセットとなる。

1. 訓練コーパスから初期モデルを作成する。
  - (a) 接尾辞配列 (suffix array) でコーパスの部分文字列を獲得し、サブワード候補とする。
  - (b) 各サブワードには、訓練コーパスの相対頻度を尤度として付与する。
2. モデルの語彙が所定の語彙サイズになるまで、以下を繰り返す。
  - (a) EM アルゴリズムでモデルの尤度を更新する。(2 回ループ)
    - i. **E ステップ:** コーパスの文を現モデルで解析し、前向き後ろ向きアルゴリズムで各サブワードの尤度を算出。
    - ii. **M ステップ:** コーパス全体でサブワードの尤度を集計し、モデルを更新。
  - (b) 低尤度のサブワードをモデルから削除する (たとえば全体の 20%)

## 3 既存モデルへのサブワード追加

新規言語が未知の文字を使用している場合、無損失符号化では、文全体が未知語になってしまい、事

前学習モデルで扱うべき語彙がわからない。対策として、文字をトークンとして扱う方法があるが、その方法ではトークン列が長くなり、翻訳などの下流タスクには不利になる。本稿ではサブワードを使うことで、語彙を適切に制限しつつ、トークン列の長さも長くしすぎないようにする。

今回は、既存コーパスの解析結果を変更せずに、新規言語のみ適切にサブワード列に符号化するため、既存モデルでは未知となる「文字」を含むサブワードだけを追加する。SentencePiece の追加学習と通常のユニグラムモデル学習との差異は、以下の 2 点である (詳細は付録参照)。

- 1.(a) 訓練コーパスから初期モデルを作成する際、既存モデルには含まれない文字から始まる部分文字列のみをサブワード候補とする。
- 2.(a) EM アルゴリズムでモデルの尤度を更新する際、コーパスを既存モデルと追加モデルの統合語彙で解析する。尤度の更新は追加モデルのみに対して行い、既存モデルは更新しない。

## 4 イヌイット語翻訳実験

本稿では、WMT-20 [10] のニュース翻訳タスクで実施された英語 (En)・イヌイット語 (Iu) 翻訳で評価する。イヌイット語の例を表 1 に示す。イヌイット語は、ユニコードの「統合カナダ先住民音節 (U+1400 - U+167F)」で記述されている。

一方、本稿で対象としている mBART-50 の SentencePiece モデルは、100 言語のコーパスから学習されたサブワード 25 万語から成り立っているが、これには統合カナダ先住民音節を含んでいない。そのため、イヌイット語をこのモデルで解析すると、ほとんどが未知語となり、サブワード化されない<sup>1)</sup>。

1) SentencePiece は、未知文字が連続すると、それらをまとめて 1 トークンとして出力する仕様となっている。そのため、イヌイット語を解析すると、結果的にスペース区切りと同じ出力となる。

表 1 英語・イヌイット語対訳文の例

言語	文
英語	Good afternoon, Nunavummiut and residents of my community.
イヌイット語	ᐅᓐᓂᓐᓂᓐᓂᓐ ᐅᓐᓂᓐᓂᓐᓂᓐ ᐅᓐᓂᓐᓂᓐᓂᓐ. ᐅᓐᓂᓐ ᐅᓐᓂᓐᓂᓐᓂᓐᓂᓐ ᐅᓐᓂᓐᓂᓐᓂᓐ.

## 4.1 実験設定

**コーパス** WMT-20 共有タスクで使用された対訳コーパスは、訓練セットが約 131 万文、開発セットが 5,173 文、テストセットが 2,971 文である。

**トークナイザー設定** 本実験では、対訳コーパスのイヌイット語側約 131 万文から、追加サブワード {2000 (2K), 4000 (4K), 8000 (8K)} を学習する。

既存のユニグラムモデルは、`spm_export_vocab` コマンドでテキストモデルにして使用、2.1 節、2.2 節を Python で実装し、3 節の変更を行った。なお、入力文字列の正規化（スペース置換を含む）は `spm_normalize` コマンドで行った。

**ベースライン** 本実験では、3 種類のベースラインのトークナイズ法を設定する。

- **mBART-50 分割モデル:** mBART-50 付属の SentencePiece モデルをそのまま用いる場合。
- **イヌイット文字分解:** mBART-50 分割モデルでトークナイズ後、イヌイット文字を含む単語を文字に分解する場合。
- **共有語彙 32K モデル:** 対訳コーパスから新規に両者の共有語彙を SentencePiece で学習する場合。語彙サイズは 3 万 2 千語とする。

**翻訳器/モデル** 本稿では、翻訳器として FairSeq [14] を使用する。

ベースとした事前学習モデル mBART-50 はエンコーダー・デコーダー型のモデルであり、対訳コーパスでファインチューニングすると翻訳器として利用できる。mBART-50 自体は多言語モデルであるので、1 回のファインチューニングで、En → Iu と Iu → En の双方向モデルを学習・使用する。

mBART-50 自体の新規言語への拡張は、単語埋込を拡張する [6] の方法を使用した。具体的には、mBART-50 でイヌイット語翻訳を実現するため、エンコーダーおよびデコーダーの単語埋込を以下の 2 点に関して拡張する。

- mBART-50 は、原言語と目的言語に言語タグを付加して学習・翻訳を行っているので、イヌイット

語の言語タグ (iu\_CA) を単語埋込に追加する。

- トークナイザーで追加したサブワードを単語埋込に追加する。

拡張部分はランダム初期化し、ファインチューニング時に他のパラメータとまとめて学習する。

なお、ベースラインとして、共有語彙 32K モデルでトークナイズした対訳コーパスだけを用いて学習した、Transformer Big の双方向モデルを用いた。

**学習/テスト** ハイパーパラメータの詳細は付録に示す。

## 4.2 トークナイズ結果

表 2 は、対訳テストセットのイヌイット語をトークナイズした結果である。テストセット 1 文あたりのトークン数と、mBART-50 モデルの語彙（単語埋込）からみた未知語率を示す。すべてのトークナイザーにおいて、高い未知語率を示しているが、これは mBART-50 に語彙を追加することで低くすることができる。表 2 の「未知語をなくすための追加語彙数」は、ゼロにするために追加する語彙数である。

ニューラルモデルでは、1 文あたりのトークン数が少ない方が解析や翻訳精度が高い場合が多いが、最も少ないのは mBART-50 分割モデルである。これは、SentencePiece が未知文字に遭遇し、スペース区切りでトークナイズしたため、全トークンの 40% 以上が未知語となる。もし、このトークナイザーで翻訳モデルを学習しようとする、150 万語以上を追加しなければならず、学習は非現実的である。

共有語彙 32K モデルも、1 文あたりのトークン数が少ないが、この語彙は mBART-50 事前学習モデルの語彙と異なるため、事前学習モデルに適用するためには、3 万 2 千語のうち、約 2 万 7 千語を追加しなければならない。

イヌイット語文字分割モデルは 141 語を追加するだけで良いが、1 文あたりのトークン数が 80 になり、入出力が長くなる。

一方、本稿の方式は、1 文あたりのトークン数は mBART-50 分割モデルや共有語彙 32K モデルより長くなっているが、イヌイット文字分解の半分程度にすることができた。本稿の提案方式は、追加語数を制御しながら未知語をなくすことができる。

## 4.3 翻訳実験結果

次に翻訳実験の結果を表 3 に示す。ベースラインとしては、Transformer Big モデル [15] を使用した。

表2 イヌイット語の1文あたりのトークン数と mBART-50 モデルの語彙から見た未知語率、および追加すべき語彙数

タイプ	トークナイザー	トークン数/文 (テストセット)	未知語率 (テストセット)	未知語をなくすための 追加語彙数 (訓練セット)
ベースライン	mBART-50 分割モデル	20.9	43.1%	1,553,466
	イヌイット文字分解	80.0	85.1%	141
	共有語彙 32K モデル	22.0	82.7%	26,657
提案方法	語彙追加 2K	37.9	68.7%	2,001
	語彙追加 4K	34.6	65.7%	4,001
	語彙追加 8K	31.8	62.8%	8,001

表3 英語・イヌイット語間翻訳 BLEU スコア。

ボールド数値は、各翻訳方向での BLEU 最高値を表す。(+) は Transformer Big モデル (共有語彙 32K モデル) に比べて有意に高く、(-) は有意に低いことを表す。(†) は mBART-50 モデル (イヌイット文字分解) に比べて有意に高いことを表す。

翻訳モデル	トークナイザー	追加埋込数	未知語率	BLEU	
				En → Iu	Iu → En
Transformer Big モデル	共有語彙 32K モデル	—	0.0%	8.3	21.5
mBART-50 モデル	mBART-50 分割モデル	0	43.1%	1.7 (-)	7.0 (-)
		1,553,466	12.1%	N/A	N/A
	イヌイット文字分解	140	0.0%	9.8 (+)	<b>23.5 (+)</b>
	共有語彙 32K モデル	26,657	0.0%	9.6 (+)	22.8 (+)
	語彙追加 2K	2,000	0.0%	<b>10.2 (+†)</b>	<b>23.5 (+)</b>
	語彙追加 4K	4,000	0.0%	9.7 (+)	23.3 (+)
	語彙追加 8K	8,000	0.0%	9.8 (+)	23.2 (+)

mBART-50 モデルは、追加埋込数の単語埋込を拡張してからファインチューニングしている。未知語率は単語埋込拡張後のテストセット未知語率である。BLEU [16] の評価には sacreBLEU [17] を使用した。有意差検定はブートストラップ再サンプリングを用い、危険率 5%とした ( $p < 0.05$ )。

まず、mBART-50 モデルを適用することにより、Transformer Big モデルに比べて、En-Iu, Iu-En どちらも、BLEU スコアを有意に向上させることができた (mBART-50 分割モデルを除く)。mBART-50 にはイヌイット語が含まれていないにもかかわらず、事前学習は有効であった。

mBART-50 翻訳モデル内の各トークナイザーに着目すると、mBART-50 分割モデルは、未知語率が高く、意味のある翻訳はできなかった。共有語彙 32K の場合は、Transformer Big モデルより BLEU が向上したが、イヌイット語文字分解よりは低かった。イヌイット語文字分解は、ベースライントークナイザーの中では、最も BLEU スコアが高かった。

イヌイット語文字分解と比較すると、我々のトークナイザーでは En-Iu の語彙追加 2K 以外では BLEU スコアに有意な差はなかった。また、サブワード数が少ない方が BLEU スコアが高い傾向は得られた。

これは、ファインチューニングがうまくトークナイザーの差を吸収したためと考えている。

まとめると、サブワード数による翻訳品質に対する明確な優位性は得られなかったが、イヌイット語をサブワード分割することで mBART-50 が翻訳に適用でき、事前学習モデルの効果が得られることを確認した。

## 5 まとめ

本稿では、多言語事前学習モデルを新規言語に対応させるため、SentencePiece トークナイザーに新規のサブワードを追加した。提案方法は、既存のモデルを変更せずに、未知の文字を含むサブワードをその尤度とともに追加した。

実験はイヌイット語に対して行い、新規言語のサブワード分割が可能になることを確認した。サブワード数は制御可能である。また、イヌイット語翻訳に事前学習モデル mBART-50 を適用し、多言語事前学習モデルを用いることで、新規言語でも BLEU スコアが向上することを確認した。

本稿の追加サブワード学習の実装は GitHub で公開している。<sup>2)</sup>

2) [https://github.com/kenji-imamura/sentpiece\\_mimic](https://github.com/kenji-imamura/sentpiece_mimic)



## 謝辞

本件は、総務省の「ICT 重点技術の研究開発プロジェクト (JPMI00316)」における「多言語翻訳技術の高度化に関する研究開発」による委託を受けて実施した研究開発による成果です。

## 参考文献

- [1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.
- [2] 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. **arXiv e-Print**, 1911.02116, 2020.
- [3] Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for neural machine translation. **Transactions of the Association for Computational Linguistics**, Vol. 8, pp. 726–742, 2020.
- [4] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 7871–7880, Online, July 2020. Association for Computational Linguistics.
- [5] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mT5: A massively multilingual pre-trained text-to-text transformer. In **Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 483–498, Online, June 2021. Association for Computational Linguistics.
- [6] Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. Extending multilingual BERT to low-resource languages. In **Findings of the Association for Computational Linguistics: EMNLP 2020**, pp. 2649–2656, Online, November 2020. Association for Computational Linguistics.
- [7] Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. Multilingual translation with extensible multilingual pre-training and finetuning. **arXiv e-Print**, 2008.00401, 2020.
- [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**, pp. 66–71, Brussels, Belgium, November 2018. Association for Computational Linguistics.
- [9] 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)**, pp. 66–75, Melbourne, Australia, July 2018. Association for Computational Linguistics.
- [10] Loïc Barrault, Ondřej Bojar, Fethi Bougares, Rajen Chatterjee, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Alexander Fraser, Yvette Graham, Paco Guzman, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, André Martins, Makoto Morishita, Christof Monz, Masaaki Nagata, Toshiaki Nakazawa, and Matteo Negri, editors. **Proceedings of the Fifth Conference on Machine Translation**, Online, November 2020. Association for Computational Linguistics.
- [11] 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)**, pp. 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics.
- [12] Christopher D. Manning and Hinrich Schütze. **Foundations of Statistical Natural Language Processing**. The MIT Press, 1999.
- [13] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying conditional random fields to Japanese morphological analysis. In **Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing**, pp. 230–237, Barcelona, Spain, July 2004. Association for Computational Linguistics.
- [14] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)**, pp. 48–53, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.
- [15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. **CoRR**, Vol. abs/1706.03762, , 2017.
- [16] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In **Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics**, pp. 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.
- [17] Matt Post. A call for clarity in reporting BLEU scores. In **Proceedings of the Third Conference on Machine Translation: Research Papers**, pp. 186–191, Brussels, Belgium, October 2018. Association for Computational Linguistics.

---

**Algorithm 1** Additional Subword Model Learning

---

**Input:** Corpus  $C$ , vocabulary size  $V$ , original model  $M_{org}$ **Output:** Additional model  $M$ 

```
1:  $M \leftarrow \text{GENERATEINITMODEL}(C)$ 
2: while  $|M| > V$  do
3:    $M \leftarrow \text{UPDATELIKELIHOOD}(M, M_{org}, C)$ 
4:    $M \leftarrow \text{UPDATELIKELIHOOD}(M, M_{org}, C)$ 
5:   Eliminate low-likelihood subwords  $\langle sw, l \rangle$  from  $M$ .
6: end while
7: return  $M$ 

8: function  $\text{GENERATEINITMODEL}(C)$ 
9:    $M \leftarrow \emptyset$ 
10:  for all substring  $sw \in C$  do
11:    Compute the likelihood  $l$  of  $sw$  from the relative frequency in  $C$ .
12:     $M \leftarrow M \cup \langle sw, l \rangle$ 
13:  end for
14:  return  $M$ 
15: end function

16: function  $\text{UPDATELIKELIHOOD}(M, M_{org}, C)$ 
17:  for all sentence  $S \in C$  do ▷ E step
18:    Analyze  $S$  using  $M \cup M_{org}$ , and compute  $l$  of  $sw \in S$  using the forward-backward algorithm.
19:  end for
20:  Compute the new  $l$  by summing all  $\langle sw, l \rangle$  in  $C$ , and update the likelihoods of  $M$ . ▷ M step
21:  return  $M$ 
22: end function
```

---

## A サブワード追加アルゴリズム

本稿のサブワード追加モデルの学習アルゴリズムをアルゴリズム 1 に示す。このアルゴリズムは元モデル  $M_{org}$  に含まれないサブワードを集めた追加モデル  $M$  を学習する。

- 行 1 では、`GENERATEINITMODEL` 関数 (line 8 - 15) で初期モデルを作成する。初期モデルには、コーパス  $C$  に含まれる部分文字列がサブワード候補として含まれる。

`SentencePiece` の通常のユニグラムモデルの学習では、すべての部分文字列を対象としていたが、追加学習では、既存モデルには含まれない文字から始まる部分文字列のみをサブワード候補とする。

- 行 2 - 行 6 がメインループで、追加モデルの尤度だけを EM アルゴリズムで 2 回更新 (`UPDATELIKELIHOOD` 関数) する。なお、解析は既存モデルと追加モデルの統合語彙で行う。そして、所定の語彙サイズになるまで低尤度のサブワードを削除する。本稿では尤度の下位 20% を

**表 4** 学習/テスト時の設定一覧

---

学習/テスト	設定
学習時	LR=0.00008, dropout=0.3, バッチサイズは 8K tokens/update, warmup=約 1 epoch, cooldown=inverse_sqrt, optimizer=adam, criterion=label_smoothed_cross_entropy
テスト時	ビーム幅は 10, 長さペナルティは 1.0

---

削除した。

## B ハイパーパラメータ

表 4 は、翻訳モデル学習 (ファインチューニング) 時、およびテスト時のハイパーパラメータ設定の一覧である。トークナイザーによって訓練データの総トークン数が変わるため、warmup は約 1 エポックで統一した。

なお、Transformer big モデル学習時の設定は、LR=0.0004、warmup=約 5 エポックにした以外は、mBART-50 モデルファインチューニング時と同じである。