自動獲得語彙に基づく多言語対応テキストフィルター

今村 賢治　　内山将夫
国立研究開発法人　情報通信研究機構
{kenji.imamura,mutiyama}@nict.go.jp

概要

本稿では、多言語文フィルターを提案する。提案方式は単言語コーパスの語彙を集計し、入力との照合を行うだけのシンプルな方式であるが、コード混在テキストの排除に効果が高い。既存の言語識別と併用することで、対象言語におけるコーパスの純度を向上させることができるため、機械翻訳用対訳コーパスに適用すると、翻訳品質を向上させることができる。また、漸進的に言語を追加でき、学習済み言語を再学習する必要がない。

1　はじめに

ニューラルモデルは、複数の言語を一つのモデルで扱うことができるため [1]、多言語モデルが一般的になってきている。しかし、それらの学習に用いる多言語コーパスは、多くの場合 Web 等から自動獲得しているため、どうしてもノイズが含まれる。コーパスから対象以外の言語を排除するためには、言語識別が有効である。しかし、既存の方式は、文の一部に他言語が混在したテキスト（これをコード混在テキストと呼ぶ）でも、その割合が少ない場合は、排除に失敗することも多い。

本稿では、単言語フィルタリング法を提案する。本稿で提案する方式は、自動獲得した語彙との照合を行うだけの、非常にシンプルな方法で、トークン単位の識別を行う。本稿では、これを対訳コーパスに対して適用し、機械翻訳の翻訳品質が向上することを示す。

本稿の方法は、コード混在テキストの削除に特に有効である。コード混在テキストが減少したコーパスをモデル学習に使用することにより、より統一された言語を生成することが可能となり、機械翻訳品質が向上する。さらに、新規言語を追加するには、単言語コーパスをトークナイズし、集計するだけなので、学習済み言語を再学習する必要はなく、漸進的に言語を追加することができる。

2　関連研究：言語識別

単言語のコーパスフィルタリングでは、言語識別が重要な要素である。

表 1 は、言語識別をまとめたものである。提案されているほぼすべての方式は、学習ベースのものである。言語識別器は、(1) 文・テキスト単位の識別と、(2) トークン・文字単位の識別に大別される[5)]。

文・テキスト単位の言語識別器を使った場合、コード混在テキストでは、その量次第では対象言語として識別されてしまう。一方、トークン単位の識別はトークン単位で言語を識別する。トークン単位の結果を元に、文のフィルタリングを決定する処理が必要であるが、判定の仕方次第でコード混在テキストをフィルタリング可能である。本稿の提案方法は、トークン単位の識別方法であるので、コード混在テキストのフィルタイングに有効である。

3　提案方式

本稿の提案方式は、言語ごとに構築した二値分類器である。単言語コーパスから語彙を自動獲得し、その語彙と入力をサブワードレベルで照合することによって、トークン単位の識別を行い、フィルタリングする。訓練およびフィルタリングは、以下の手順で行う。

訓練

1. 単言語コーパスをサブワード分割する。単言語コーパスは対象言語のみを含んでいることが望ましいが、後述のように、ある程度のノイズは許容できるので、Web テキストを他の言語識別で自動収集したものでよい。また、トークナイ

1）https://fasttext.cc/docs/en/language-identification.html
2）https://github.com/saffsd/langid.py
3）https://github.com/google/cld3
4）https://github.com/davidjurgens/equilid
5）トークン単位の識別器は、入手が困難だったり、実装が古くて動作しなかったりして、動作確認できなかった。

表 1　言語識別のまとめ

単位	名称	言語数	説明	備考
文・テキスト	fastText[1)]	176	skip-gram [2] ベースの多クラス分類器	[3, 4, 5]
	langid.py[2)]	97	決定性有限オートマトン (DFA) ベースの naïve Bayes 分類器	[6]
トークン・文字	CMX	100	フィードフォワードネットワーク (FFN) の多クラス分類器、トークン単位で判定	[7]
	CLD3[3)]	107	文字 n-gram 入力を平均化する、FFN の多クラス分類器	Google Chrome browser plugin.
	LanideNN	131	文字埋込を双方向再帰型ニューラルネット (RNN) 分類器で文字単位に識別	[8]
	Equilid[4)]	70	トークン単位で判定する 3 層ニューラルネット	[9]

ザーはコーパスから教師なし学習できるものを使用する。本稿では、SentencePiece [10] を使用する。

2. サブワードトークンを集計し、出現頻度順にソートして語彙として保存する。

フィルタリング

3. ステップ 2. で作成した語彙を読み込む。その際、カバレッジが語彙制限 VL になるように、出現頻度上位のサブワードのみを選択し、それを有効語彙 V とする。出現頻度下位のサブワードはノイズである場合が多いので、これは排除される。
4. フィルタリング対象のノイズ入りコーパスを、ステップ 1 で使用したものと同じトークナイザーでサブワード分割する。
5. トークンと有効語彙を照合する。
6. 各文の有効語彙の割合が閾値（有効トークン率 TR）未満のものは、ノイズとして排除し、それ以外を出力する。具体的には、以下のフィルター関数 vocabFilter(W) の条件を満たす文 W をフィルタリング結果として出力する。

$$\begin{aligned} \text{vocabFilter}(W) &= \frac{\sum_i \text{match}(w_i)}{|W|} \\ &\geq TR, \\ \text{match}(w) &= \begin{cases} 1 & \text{if } w \in V \\ 0 & \text{else} \end{cases}, \end{aligned}$$

ただし、W は判定対象の文、w_i は W の i 番目のトークンである。

この方式には、以下の特徴がある。

- 全体構成について、トークン単位の識別を集計することでフィルタリングするので、有効トークン率 TR を高めに設定することによって、コード混在テキストも適切にフィルタリングできる。
- 提案方式は、言語ごとの二値分類器である。多クラス分類器の場合、言語を後から追加するには、（最悪）全言語を再学習する必要があるが、提案方式は対象言語の二値分類器を構築するだけで、他の言語を作り直す必要はなく、言語を漸進的に追加可能である。
- 有効語彙について、語彙獲得用の単言語コーパスにノイズが含まれていても、上位には機能語、中間に内容語、下位にノイズが集まるため、言語識別が可能である。（表 2）。
- Unigram 言語モデルによるフィルタリングと類似しているが、言語モデルは正しい文であっても、内容語の出現確率はどうしても小さくなる。提案方式はサブワード単位の識別であるため、単語そのものが有効語彙に含まれなくても、サブワードが含まれる場合があり、内容語の識別にも有利である。

有効語彙　後述の第 4 節の実験設定で獲得した、ドイツ語の有効語彙の一部を、表 2 に示す。総語彙数は言語によって著しく異なっていたが、共通していることは、上位に各言語の大部分を構成する記号、機能語が占めていることと、最下位はほぼ他の言語が混入したノイズであるということである。したがって、下位を無効にすることで、各言語の有効語彙が得られる。

語彙制限 VL は、語彙の有効・無効を区別する閾値（ボーダー）であるが、これは、累積カバー率をベースに決定した（今回は 99.5%）。有効語彙数は、英語は 20,927 語、ドイツ語は 21,342 語、パシュトー語は 6,210 語、クメール語は 12,602 語となった。ボーダー前後は、各言語の内容語のサブワードになっており、これを有効とするか、無効とするかは、決定的な閾値では不正確である。そのため、有効トークン率 TR を緩める（本稿では 0.9）ことで、

表 2　ドイツ語の語彙の例。第 4 節の実験で獲得した語彙。累積カバー率 99.5%までのサブワードを有効語彙とした。

順位	サブワード	頻度	累積カバー率	
1	.	597M	3.71%	有効語彙
2	,	566M	7.23%	
3	_und	276M	8.95%	
4	_die	245M	10.47%	
5	en	243M	11.98%	
:				
21340	Wikipedia	13,937	99.49%	
21341	teis	13,936	99.49%	
21342	_oxid	13,936	99.49%	
21343	_Malo	13,935	99.50%	
21344	408	13,935	99.50%	
21345	ARS	13,934	99.50%	
:				
110927	Ấ	1	100.00%	
110928	높	1	100.00%	
110929	받	1	100.00%	

表 3　実験で使用した対訳コーパス

言語対	ノイズ入りコーパス 文数	ノイズ入りコーパス トークン数	選択後コーパス 文数	選択後コーパス トークン数
De-En	104M	1.0B	-	100M
Ps-En	1.02M	11M	-	5.0M
Km-En	4.17M	58M	-	5.0M

誤差を許容可能としている。

4　実験

本稿では、提案方式を vocabFilter と呼称し、フィルタリングコーパスを使った機械翻訳品質で評価する。

4.1　実験設定

vocabFilter の設定　語彙獲得のための単言語コーパスには、CC-100 [11] を使用した。

トークナイザーは SentencePiece [10] を使用した。トークナイザーのモデルは、事前学習モデル mBART [12] および XLM-R [11] で使われているものを流用した。このモデルは、100 言語をサポートし、のべ 250K のサブワードを語彙として持っており、[13] で未知語 (UNK) が少ないと報告されている。また、ハイパーパラメータは、$VL = 0.995, TR = 0.9$ とした。

対訳コーパスフィルタリングタスク　本稿では、Conference on Machine Translation(WMT) の Parallel corpus filtering 共有タスク [14, 15] に準じて評価を行う。このタスクは、主催者からノイズ入り対訳コーパスが提供されている。このうち、文アライメントスコアが提供されている 2018 年のドイツ語・英語 (De-En、高リソース言語である)、2020 年のパシュトー語 (Ps-En、低リソース言語)、クメール語 (Km-En、低リソース言語) で評価した[6]。(表 3)。

評価は以下の手順で行った。

1. ノイズ入り対訳コーパスを提案方式等でフィルタリング。
2. フィルタリング結果をアライメントスコアでソートし、上位から一定トークン数（英語基準）の文を選択した。ドイツ語は 1 億トークン、パシュトー語、クメール語は 500 万トークンである（表 3)。したがって、対訳コーパスに含まれる情報量は、フィルタリング方式によらず一定とみなす。
3. 翻訳モデルを学習。翻訳器は FairSeq [16] を使用した。ハイパーパラメータの詳細は付録 A に示す。
4. テストセットの翻訳品質で評価した。テストセットは WMT 提供のものを使用し、ドイツ語は devtest セット、パシュトー語とクメール語は devtest, test 両方を結合して使用した。翻訳品質は Flores-200[17, 18] 用トークナイザーを使った sacreBLEU [19] で評価した。今回は、表層がうまく訳されていることが重要なので、BLEU を使用した。

比較方式　WMT-2020 共有タスクのベースラインと同様に、fastText によるフィルタリングをベースラインとして、提案方式を組み合わせる。また、原言語・目的言語別に適用する。

4.2　結果

4.2.1　翻訳品質

表 4 は、各フィルターを、原言語・目的言語に適用したときの BLEU スコアである。

提案方式 vocabFilter の効果は、言語によって異なる。高リソース言語であるドイツ語（De ↔ En）は、vocabFilter 単体では fastText 単体より翻訳品質は落ちたが、両者を併用することにより BLEU スコアは有意に向上した。

低リソース言語のうち、パシュトー語（Ps ↔ En）

6)　2019 年は文アライメントスコアが提供されていないため、本評価から除外した。

表 4　各言語識別を適用したときの BLEU スコア。太字はその言語方向の最高値、下線は次点。(+) と (-) マークは、それぞれ fastText のみ（最上位行）と比較したとき、有意に向上・悪化したことを表す（$p < 0.05$, bootstrap resampling を使用）。

fastText		vocabFilter		XX→En			En→XX		
原言語	目的言語	原言語	目的言語	De→En	Ps→En	Km→En	En→De	En→Ps	En→Km
✓	✓			29.5	8.8	7.3	27.5	10.7	14.8
		✓	✓	27.7 (-)	8.6	**8.1** (+)	25.4 (-)	10.6	**15.1** (+)
✓	✓	✓		**30.9** (+)	**9.1** (+)	**8.1** (+)	**28.6** (+)	**11.0** (+)	**15.1** (+)
✓	✓		✓	30.2 (+)	<u>8.9</u>	**8.1** (+)	28.1 (+)	**11.0** (+)	14.9
✓	✓	✓	✓	<u>30.8</u> (+)	8.8	7.7 (+)	<u>28.4</u> (+)	10.7	14.2 (-)

表 5　fastText が OK となった文のうち、vocabFilter が排除した例

言語	No.	例文（トークナイズ＋照合結果）	有効トークン率
英語	1	▁" Ma hl zeit ", ▁cho reo graph er , ▁Theater ▁am ▁Wall , ▁War endorf	0.875
	2	▁ -▁ ي رخصت▁ ملي▁Em ba ssy ▁of ▁Afghanistan ▁in ▁Ott awa	0.750
	3	▁ FOL LOW ▁US ▁ON ▁SOCIAL !	0.857
ドイツ語	4	▁Kontakt ▁B Berlin 1 ▁2015 -08- 26 T 17 :00 :26 +00:00	0.833
	5	▁T sche chi en , ▁Li šov	0.857
	6	▁Homepage ▁ | ▁Druck en ▁ | ▁Nach ▁oben	0.778
パシュトー語	7	▁ 446 #▁ 5▁ وړاندې▁ ورځې▁ و مياشت▁ 7▁by ▁Dari us s ssss	0.692
	8	5- رح ▁ امريكا▁ شمالي▁C ▁▁ د▁ اور پکFSX ▁& ▁P 3 D ▁2.5	0.867
	9	▁Chang zhou ▁Daily s ▁▁ الت مح صو▁ پام لر ني▁ دCo . ،▁Ltd	0.800
クメール語	10	▁ផ្នែក ▁Put z meister ▁(▁25 ▁)	0.857
	11	▁English , ▁У к р а ї н с ь к а , ▁Français , ▁Español ... ▁សូម ▁មើល ▁ ផ្នែក ▁ទ ន់ ▁បន្ថែម ▁ទៀត	0.800
	12	▁Ma un fac turer ▁ ស្បែក ជើង បុរស សុវត្ថិភាព ក្រៅ	0.800

は、ドイツ語に比べて効果は少ないが、fastText に加えて vocabFilter をソース側に適用することにより、翻訳品質は向上した。

一方、クメール語（Km ↔ En）については、En → Km の全適用以外は、vocabFilter 単体、併用ともに、ベースラインより BLEU スコアが向上している。

このように、言語によって効果が異なるが、fastText に vocabFilter を併用することにより、翻訳品質が向上する傾向があった。

4.2.2　vocabFilter が排除する文の例

表 5 は、fastText で OK となった文のうち、vocabFilter が排除した文の例である。文は SentencePiece によりトークナイズしてあり、赤字は語彙との照合に失敗したトークンである。有効トークン率は、有効トークンの割合で、これが 0.9 未満の文は排除される。

この例を見ると、言語によらず、fastText で OK となった大部分のコード混在テキストは適切に排除している。ただし、数字に関しては、長い数字トークンは語彙に含まれていないことが多く、識別結果が NG になることも多い (No. 4, 7, 8 参照)。また、正しい対象言語でも、一部は語彙に一致しない場合もある (No. 3, 5)。

このように、すべての文を正しくフィルタリングできるわけではないが、コード混在文を自動で排除できるメリットは大きいと考える。

5　おわりに

本稿では、シンプルな単言語フィルタリング法を提案した。提案方式は、自動獲得した語彙と入力トークンの照合を行うだけの二値分類器であるが、サブワード単位で行うため、未知の単語も比較的照合可能である。また、新規言語を追加するには、単言語コーパスをトークナイズして集計するだけなので、既存言語のモデルを再学習する必要がなく、漸進的に言語を増やすことができる。

これをアライメントスコアベースのフィルタリングが施されたコーパスに対して適用し、機械翻訳の翻訳品質が向上することを示した。提案方式は、とくにコード混在テキストの削除に有効である。他の言語識別と併用することにより、より純粋な言語コーパスを得ることができる。本方式でフィルタリングしたコーパスを用いて学習した多言語モデルは、みんなの自動翻訳 ®[7] の “日常会話” という名前で公開している。また、フィルタリングプログラムは獲得済み語彙を含めて公開を検討している。

7） https://mt-auto-minhon-mlt.ucri.jgn-x.jp/

参考文献

[1] Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's multilingual neural machine translation system: Enabling zero-shot translation. **Transactions of the Association for Computational Linguistics**, Vol. 5, pp. 339–351, 2017.

[2] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, **Advances in Neural Information Processing Systems**, Vol. 26. Curran Associates, Inc., 2013.

[3] Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. Learning word vectors for 157 languages. **arXiv e-print**, 1802.06893, 2018.

[4] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. **arXiv e-print**, 1607.01759, 2016.

[5] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. **arXiv e-print**, 1607.04606, 2017.

[6] Marco Lui and Timothy Baldwin. langid.py: An off-the-shelf language identification tool. In **Proceedings of the ACL 2012 System Demonstrations**, pp. 25–30, July 2012.

[7] Yuan Zhang, Jason Riesa, Daniel Gillick, Anton Bakalov, Jason Baldridge, and David Weiss. A fast, compact, accurate model for language identification of codemixed text. In **Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing**, pp. 328–337, October-November 2018.

[8] Tom Kocmi and Ondřej Bojar. LanideNN: Multilingual language identification on character window. **arXiv e-print**, 1701.03338, 2017.

[9] David Jurgens, Yulia Tsvetkov, and Dan Jurafsky. Incorporating dialectal variability for socially equitable language identification. In **Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)**, pp. 51–57, July 2017.

[10] 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, November 2018.

[11] 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**, pp. 8440–8451, July 2020.

[12] 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.

[13] Kenji Imamura and Masao Utiyama. An empirical study of multilingual vocabulary for neural machine translation models. In **Proceedings of the Eleventh Workshop on Asian Translation (WAT 2024)**, pp. 22–35, November 2024.

[14] Philipp Koehn, Huda Khayrallah, Kenneth Heafield, and Mikel L. Forcada. Findings of the WMT 2018 shared task on parallel corpus filtering. In **Proceedings of the Third Conference on Machine Translation: Shared Task Papers**, pp. 726–739, October 2018.

[15] Philipp Koehn, Vishrav Chaudhary, Ahmed El-Kishky, Naman Goyal, Peng-Jen Chen, and Francisco Guzmán. Findings of the WMT 2020 shared task on parallel corpus filtering and alignment. In **Proceedings of the Fifth Conference on Machine Translation**, pp. 726–742, November 2020.

[16] 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, June 2019.

[17] NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. No language left behind: Scaling human-centered machine translation. **arXiv e-print**, 2207.04672, 2022.

[18] Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. **Transactions of the Association for Computational Linguistics**, Vol. 10, pp. 522–538, 2022.

[19] Matt Post. A call for clarity in reporting BLEU scores. In **Proceedings of the Third Conference on Machine Translation: Research Papers**, pp. 186–191, October 2018.

A フィルタリング実験時の翻訳器のハイパーパラメータ

4.1 節のフィルタリング実験では、表 6 に示すハイパーパラメータで翻訳モデルを学習し、翻訳実験を行った。

表 6 翻訳モデル学習/翻訳時のハイパーパラメータ

モデル構造	
Architecture	Transformer
# of layers	5
Embedding dimension	512
FFN inner dimension	2,048
Attention heads	2
Other model settings	Share all embeddings Normalize before
訓練	
Dropout	0.4
Attention dropout	0.2
ReLU dropout	0.2
Loss function	Label smoothed cross-entropy
Label smoothing	$\epsilon = 0.2$
Optimizer	Adam ($\beta_1 = 0.9, \beta_2 = 0.98$)
Learning rate	1e-3
LR scheduler	Inverse square root
Warm-up steps	4,000
Global batch size	Roughly 16,000 tokens
Training epochs	100
翻訳	
Beam width	5
Length penalty	1.2