単語の長さと構成要素を考慮した単語レベルの摂動

平岡 達也 [1]　高瀬 翔 [1]
内海 慶 [2]　欅 惇志 [2]　岡崎 直観 [1]
[1] 東京工業大学　[2] デンソーアイティーラボラトリ
{tatsuya.hiraoka@nlp., sho.takase@nlp., okazaki@}c.titech.ac.jp
{kuchiumi, akeyaki}@d-itlab.co.jp

概要

本稿では，単語置換による摂動のシンプルな改善手法として，単語の長さを考慮した単語置換（WR-L）と，単語の構成要素を考慮した単語置換（CWR）を提案する．WR-L では，置換対象の単語の長さをもとにしたポアソン分布から置換後の単語を選択する．CWR では，置換対象の単語の構成要素（部分文字列・重複文字列）から置換後の単語を選択する．評価実験により，WR-L と CWR が文書分類と機械翻訳の性能向上に寄与することを示す．

1　はじめに

単語置換による単語レベルの摂動は，自然言語処理で広く用いられる [1, 2]．一般的な単語置換 [3, 4] では，入力文中の単語を語彙に含まれるランダムな単語に置換する．これは単語を一様分布からサンプリングする単純な手法であるが，敵対的摂動のような複雑な摂動と同程度に効果的であることが知られている [2]．しかし，従来の単語置換では置換対象の単語とは無関係な単語を頻繁に選択することになる．そのため，摂動を加える単語の割合を高く設定すると，文中の殆どの単語が無関係な単語に置換されてしまい，モデルの性能に悪影響を及ぼす．摂動を活用するためには，何度も試行を繰り返してハイパーパラメータを慎重に設定する必要がある．

単語レベルの摂動の異なるアプローチとして，サブワード正則化 [5, 6, 7] が挙げられる．サブワード正則化では，事前に作成した言語モデルに基づいて，学習エポックごとに異なる単語分割をサンプリングしてモデルの学習を行う．サブワード正則化は単語分割をサンプリングするため，元の文と無関係な単語は採用されない．しかし，単語分割のサンプリングには複雑な処理を要するため，単語置換に比べると摂動の処理に時間がかかるという問題点がある．また，サブワード正則化は単語置換と比較して，性能向上が限定的となることがある．

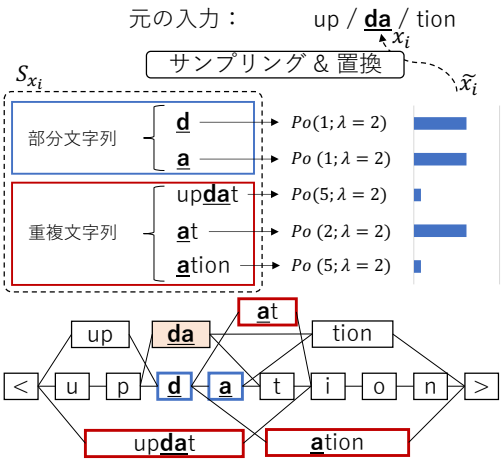

図 1: “up/da/tion” に含まれる単語 “da” を CWR − L によって置換する場合の概要．

本稿では，単語置換とサブワード正則化の折衷案として，高速かつ効果的な単語レベルの摂動手法を提案する．提案手法では，単語置換によるサンプリングの対象となる語彙を，置換対象の単語の (1) 長さと (2) 構成要素によって制限する．単語の長さを用いたアプローチ (1) では，置換対象の単語の長さに応じて単語のサンプリングに用いる分布を重み付けする．単語の構成要素を用いたアプローチ (2) では，サンプリングの対象となる語彙を，置換対象の単語を構成する要素に制限する．これらの工夫により，置換対象の単語が無関係な単語に置き換えられることを防ぎ，ハイパーパラメータの変更に対しても頑健にタスクの性能が向上するような単語レベルの摂動を実現する．さらに，提案手法では単語分割のサンプリングを行わないため，サブワード正則化よりも高速に摂動処理を実行できる．評価実験により，提案手法が文書分類と機械翻訳において性能の向上に寄与することを確認する．

2 提案手法

提案手法の説明の前に，一般的な単語置換の概要を説明する．I 個の単語からなる文 $\boldsymbol{x} = x_1, ..., x_i, ..., x_I$ に対して，単語置換では確率 a で選択した単語 x_i を $\tilde{x}_i$ に置き換える．

$$\tilde{x}_i \sim Q_V \tag{1}$$

$$x_i = \begin{cases} \tilde{x}_i & \text{with probability} \quad a \\ x_i & \text{with probability} \quad 1-a \end{cases}. \tag{2}$$

ここで Q_V は，語彙 V の全体に対する一様分布であり，a は置換の頻度を設定するためのハイパーパラメータである．本稿では，確率 a で選択された x_i を置換対象の単語，$\tilde{x}_i$ を置換後の単語と呼ぶ．

2.1 長さを考慮した単語置換（WR-L）

従来の単語置換では，置換対象の単語の長さによらず，単語のサンプリングを行う．この問題を解決するために，置換対象の単語の長さに近い長さの単語を優先的に選択するような単語置換（Word Replacement using Length: WR-L）を提案する[1)]．WR-L では，置換対象の単語の長さが平均となるようなポアソン分布[2)]を用いて Q_V を重み付けし，以下の確率を用いて単語をサンプリングする．

$$p(\tilde{x}_i|x_i) = \frac{\text{Poisson}(L_{\tilde{x}_i}; \lambda = L_{x_i})}{Z}. \tag{3}$$

ここで L_{x_i} は単語 x_i の文字数，Z は確率の合計を 1 とするための正規化項である．

2.2 構成要素を考慮した単語置換（CWR）

従来の単語置換では一様分布を用いているため，置換対象の単語とは無関係な単語が頻繁にサンプリングされる．これを解決するために，置換対象の単語の構成要素を考慮した単語置換（Compositional Word Replacement: CWR）を提案する．

CWR では，単語のサンプリングを行う語彙 V を，置換対象の単語の構成要素（部分文字列と重複文字列）からなる S_{x_i} に制限する．具体的には，置換対象の単語を構成する部分文字列と，置換対象の単語の一部を含む重複文字列に語彙を制限する．例えば図 1 のように，“up/**da**/tion” に含まれる “da” という単語を置換対象とすると，部分文字列は “d” と “a” の 2 種類，重複文字列は “up**da**t”，“**a**t,”，‘**a**tion,” の 3 種類である．制限された語彙 S_{x_i} に対して，一様分布 $Q_{S_{x_i}}$ から置換後の単語をサンプリングする．

各単語ごとに，学習データ全体の単語分割候補を用いて重複文字列を事前に計算しておく．このとき，異なる文脈で用いられる単語であっても，単語ごとに重複文字列を共有する．例えば，単語 “da” が “up/da/tion” と “pan/da” の 2 つの文脈で用いられている場合，この単語の重複文字列は “pan/da” に含まれる “an**d**” と，“up/da/tion” に含まれる “up**da**t”，“**a**t”，“**a**tion” の合計 4 種類から構成される[3)]．

また，WR-L と CWR は同時に利用してもよい．この場合，CWR によって制限されたサンプリングのための語彙 S_{x_i} の一様分布に対して，ポアソン分布を用いた重み付けを行う．

表 1: 単語置換の例．太字は置換された単語を示す．

Method	Perturbed Example
Vanilla	_Love / _the / _updated / _format
SR	_Love / _the / **_update** / **d** / **_form** / **at**
WD	_Love / _the / **[PAD]** / _format
UTR	_Love / _the / **[UNK]** / _format
LM	_Love / _the / **_the** / _format
WR	_Love / _the / **char** / _format
WR-L	_Love / _the / **_nothing** / _format
CWR	_Love / _the / **up** / _format
CWR-L	_Love / _the / **_update** / _format

3 実験

提案手法の有効性を確かめるために，文書分類と機械翻訳で実験を行う．比較対象の手法として，単語レベルの摂動を用いない場合（**Vanilla**）と，通常の単語置換による摂動（**WR**）を用いる．さらに，他の単語レベルの摂動として以下の手法を用いる．

サブワード正則化（SR）では，各学習エポックごとに言語モデルから単語分割をサンプリングして用いる．本稿では，SentencePiece [5] を利用した．

Word Dropout (WD) では，(2) 式の $\tilde{x}_i$ として分散表現がゼロベクトルであるトークンを使用する [11]．

Unknown Token Replacement (UTR) では，(2) 式の $\tilde{x}_i$ として未知語トークンを使用する [12]．

Language Model (LM) では，ランダムに選択した単語を言語モデルに基づいてサンプリングした単語に置換する．言語モデルには SentencePiece を用い，これを利用できない設定では単語の頻度数え上げによって作成したユニグラム言語モデルを用いる．

1) 置換後の単語の長さの分布を付録図 4 に示した．
2) ポアソン分布を用いたノイズは，自然言語処理の離散的な入力を用いた学習に適していると考えられる [8]．
3) S_{x_i} を求める処理の概要を付録のアルゴリズム 1 に示した．

表 2: 文書分類での実験結果（5 回試行の平均 F1 値）．太字は手法間で最大値を示し，下線は WR を有意（マクネマー検定で $p < 0.05$）に上回ることを示す．

Dataset	Vanilla	SR	WD	UTR	LM	WR	WR-L	CWR	CWR-L
Twitter(En)	75.51	77.52	76.27	76.35	76.53	77.14	77.64	76.11	**77.79**
+ BERT	82.03	-	82.30	82.25	82.10	82.07	82.08	82.19	**82.33**
Twitter(Ja)	86.42	86.41	86.69	86.68	87.25	87.30	**87.36**	86.71	87.11
Weibo(Zh)	93.10	93.18	93.53	93.65	93.21	93.44	93.41	93.24	**93.70**
Rating(En)	65.21	65.7	66.77	65.38	66.72	67.50	**67.56**	65.42	67.01
+ BERT	71.30	-	71.68	71.47	71.54	71.83	71.65	71.84	**72.02**
Rating(Ja)	52.46	52.46	53.01	52.62	53.21	53.33	**53.39**	52.76	53.34
Rating(Zh)	48.71	49.04	48.96	48.85	49.63	49.60	**49.83**	49.13	49.71
Genre(En)	67.69	67.81	72.42	72.47	72.27	71.55	72.19	67.83	**72.76**
+ BERT	77.64	-	79.09	79.23	78.89	79.07	78.85	79.04	**79.43**
Genre(Ja)	50.42	50.03	52.07	51.92	52.17	51.82	51.85	50.64	**52.32**
Genre(Zh)	47.83	47.85	48.89	48.92	49.10	48.60	**49.83**	47.73	49.06
Average w/o BERT	65.26	65.56	66.51	66.32	66.68	66.70	**67.01**	65.51	66.98
Average w/ BERT	68.19	-	69.31	69.15	69.39	69.44	69.64	68.55	**69.72**

表 3: 機械分類での実験結果（3 回試行の平均 SacreBLEU 値 [9]）．太字は手法間で最大値を示し，下線は WR を有意（Bootstrap Resampling [10] で $p < 0.05$）に上回ることを示す．

Datasets		Vanilla	SR	WD	UTR	LM	WR	WR-L	CWR	CWR-L
IWSLT14	De-En	33.92	34.75	34.81	34.84	34.46	34.68	**34.91**	34.73	34.90
	En-De	28.02	**29.04**	28.91	28.94	28.67	28.72	28.83	28.59	28.95
IWSLT15	Vi-En	28.83	29.29	29.22	29.35	28.87	29.37	**29.63**	29.33	29.51
	En-Vi	30.39	31.55	31.32	31.42	31.52	31.04	31.29	31.57	**31.69**
	Zh-En	20.27	21.19	20.86	20.95	18.65	20.86	21.26	21.36	**21.56**
	En-Zh	14.50	15.20	15.17	15.18	14.70	15.00	15.21	15.32	**15.35**
	Average	25.99	26.84	26.72	26.78	26.15	26.61	26.86	26.82	**26.99**

提案手法である **WR-L** と **CWR** に加えて，これらを組み合わせた手法を **CWR-L** と表記する．各手法による単語置換の例を表 1 に示した．これらの摂動手法のうち，SR 以外は式 (2) で説明したハイパーパラメータ a によって単語置換の割合を制御する．SR では，言語モデルの分布を制御するハイパーパラメータ b を用いる[4]．すべてのデータセットにおいて，ハイパーパラメータの候補は 0.1 から 0.9（0.1 間隔）とし，検証データを用いて最適値を決定した．

3.1 文書分類

実験設定: 文書分類の実験では，3 言語を用いた 9 件のデータセットを利用する．Twitter(En)，Twitter(Ja)，Weibo(Zh)，はそれぞれ英語，日本語，中国語によるショートテキスト SNS での感情分析データセットである．Rating と Genre は，Amazon [13]，楽天市場 [14]，JD.com [15] のレビューデータセットから作成した英語，日本語，中国語のレート予測とジャンル予測タスクである．日本語と中国語のデータセットは，それぞれ MeCab [16] と jieba [17] で事前分割を行った．その後，すべてのデータセットについて SentencePiece で単語分割を行った．語彙の規模は，感情分析データセットで 16K，その他のデータセットで 32K である．文書分類器には BiLSTM をベースとした手法 [18] を用いた．英語のデータセットについては，HuggingFace [19] による BERT-base [20] を用いた実験も行う（+BERT）[5]．

実験結果: 表 2 に手法ごとの文書分類の性能を示した．結果より，提案手法である WR-L はベースとなっている WR を 12 件のデータセットで上回ることが示された．また，提案手法の組み合わせである CWR-L により，複数のデータセットで性能の向上が得られることも分かった．CWR-L の平均スコアは他の手法よりも高く，WR-L は CWR-L と同程度である．一方で，置換対象の単語の構成要素だけを考慮する CWR は，複数のデータセットで他の手法の性能を下回った．ここから，単語の長さを考慮する WR-L によって文書分類の性能向上が得られ，さらに単語の構成要素の考慮を組み合わせることで（CWR-L），より高い性能が得られることが示唆され

4) b は既存研究 [5] における α と同じである．

5) BERT は WordPiece を使用するため SR は利用できない．

る．ベースライン手法間の比較では，WR と LM が高い性能を示す一方で，SR は多くのデータセットで性能を向上させることができなかった．

3.2 機械翻訳

実験設定: 機械翻訳では Transformer [21] を用い，Fairseq [22] による IWSLT 設定を利用した．単語レベルの摂動は低資源での実験設定で効果的であると報告されているため [5]，IWSLT コーパスのうち，De-En，Vi-En，Zh-En ペアの双方向について実験を行う．単語分割には SentencePiece を用い，語彙の規模は各言語ごとに 32K とした．ただし，中国語については jieba による事前分割を行った．

実験結果: 表 3 に，手法ごとの機械翻訳の性能を示した．表において，SR の性能は他のベースラインに比べて高いことがわかる．CWR は SR よりも単語置換の制約が緩いが，実験結果より SR と同程度の性能に達することがわかった．さらに，WR-L も SR よりも高い性能に達しており，CWR-L は 3 ペアで最も高い性能に達している．これらの実験結果から，機械翻訳において単語分割を考慮した摂動（SR, CWR）が効果的であり，さらに単語の長さを考慮することも性能の向上に寄与することが示された．

4 分析

4.1 ハイパーパラメータの影響

ハイパーパラメータが各手法に与える影響を調べるために，3.1 節で用いたデータセットでの平均性能をハイパーパラメータの値 (a, b) ごとに測定した．図 2 より，CWR-L は多くの a の値で他の手法よりも性能が高いことが示された．WR と LM はベースライン間で高い性能を示しているが，大きい a の値を用いると性能が低下する．一般的に使用される小さい a の値において，WR-L の最高性能は WR よりも高く，LM と同程度である．ここから，LM, WR, WR-L はハイパーパラメータに敏感であり，慎重なハイパーパラメータの設定が必要であると言える．CWR は摂動を用いない場合（Vanilla）と同程度であるが，CWR と CWR-L はともに，a の値によらず比較的安定した性能を示している．これは語彙の制限によって，置換の割合が大きくても元の文の情報を失わないためである．ここから，語彙を制限する CWR によってハイパーパラメータに対する頑健性が得られ，単語の長さを考慮する（-L）ことでさらなる性能向上が得られると結論付けられる．

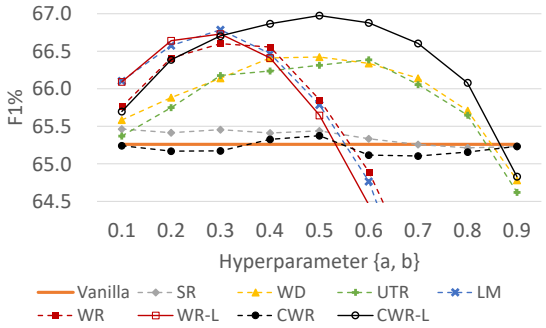

図 2: 文書分類の 9 データセット（BERT を除く）でのハイパーパラメータごとの F1 値の平均．

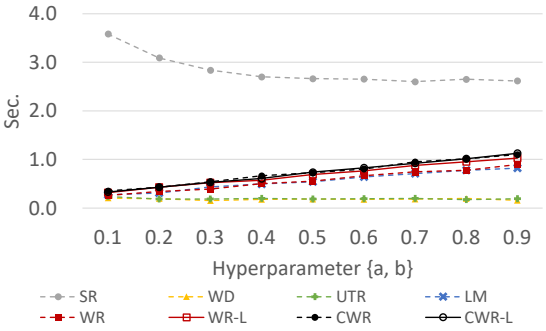

図 3: Amazon データセットの学習データにおける 10K 文あたりの処理速度（10 回試行の平均）．

4.2 処理速度

本研究では，高速で効果的な摂動の手法の開発も目的としている．本節では，Amazon データセットの学習データ（96,000 文，文平均 84.91 文字）での摂動の処理速度を比較する．

図 3 に，各手法の 10 回試行での処理速度の平均時間を示した．提案手法は語彙の制約や長さによる分布への重み付けを行うため，WR や LM よりもわずかに処理速度が低下している．一方で，明示的に単語分割のサンプリングを行う必要がないため，SR よりも高速に摂動処理を実行できる．ここから，提案手法（特に CWR-L）は性能向上と処理速度の観点から優れた摂動の手法であると結論付けられる．

5 おわりに

本稿では，高速で効果的な単語レベルの摂動手法を提案した．実験結果より，提案手法が文書分類と機械翻訳の性能向上に寄与することが分かった．CWR-L はハイパーパラメータを慎重に選ばずとも高い性能を達成しつつ，サブワード正則化よりも高速に摂動処理を実行できる優れた手法である．

謝辞

この成果は, 国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の委託業務（JPNP18002）の結果得られたものです.

参考文献

[1] Sosuke Kobayashi. Contextual augmentation: Data augmentation by words with paradigmatic relations. In **Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)**, pp. 452–457, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[2] Sho Takase and Shun Kiyono. Rethinking perturbations in encoder-decoders for fast training. In **Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 5767–5780, Online, June 2021. Association for Computational Linguistics.

[3] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In **Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 1**, pp. 1171–1179, 2015.

[4] Xiang Zhang and Yann LeCun. Text understanding from scratch. **arXiv preprint arXiv:1502.01710**, 2015.

[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)**, pp. 66–75, 2018.

[6] Tatsuya Hiraoka, Hiroyuki Shindo, and Yuji Matsumoto. Stochastic tokenization with a language model for neural text classification. In **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics**, pp. 1620–1629, 2019.

[7] Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. BPE-dropout: Simple and effective subword regularization. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 1882–1892, Online, July 2020. Association for Computational Linguistics.

[8] Masaaki Nagata. Automatic extraction of new words from Japanese texts using generalized forward-backward search. In **Conference on Empirical Methods in Natural Language Processing**, 1996.

[9] 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.

[10] Philipp Koehn. Statistical significance tests for machine translation evaluation. In **Proceedings of the 2004 conference on empirical methods in natural language processing**, pp. 388–395, 2004.

[11] Yarin Gal and Zoubin Ghahramani. A theoretically grounded application of dropout in recurrent neural networks. **Advances in neural information processing systems**, Vol. 29, pp. 1019–1027, 2016.

[12] Huaao Zhang, Shigui Qiu, Xiangyu Duan, and Min Zhang. Token drop mechanism for neural machine translation. In **Proceedings of the 28th International Conference on Computational Linguistics**, pp. 4298–4303, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics.

[13] Ruining He and Julian McAuley. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In **proceedings of the 25th international conference on world wide web**, pp. 507–517, 2016.

[14] Rakuten, Inc. Rakuten dataset. Informatics Research Data Repository, National Institute of informatics. (dataset)., 2014.

[15] Yongfeng Zhang, Min Zhang, Yi Zhang, Guokun Lai, Yiqun Liu, Honghui Zhang, and Shaoping Ma. Daily-aware personalized recommendation based on feature-level time series analysis. In **Proceedings of the 24th international conference on world wide web**, pp. 1373–1383, 2015.

[16] Taku Kudo. Mecab: Yet another part-of-speech and morphological analyzer. http://taku910.github.io/mecab/, 2006.

[17] Sun Junyi. jieba. https://github.com/fxsjy/jieba, 2013.

[18] Peng Zhou, Zhenyu Qi, Suncong Zheng, Jiaming Xu, Hongyun Bao, and Bo Xu. Text classification improved by integrating bidirectional lstm with two-dimensional max pooling. In **Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers**, pp. 3485–3495, 2016.

[19] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-art natural language processing. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations**, pp. 38–45, Online, October 2020. Association for Computational Linguistics.

[20] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018.

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. **Advances in neural information processing systems**, Vol. 30, pp. 5998–6008, 2017.

[22] 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 NAACL-HLT 2019: Demonstrations**, 2019.

表 4: 各手法・各データセットごとの実験（表 2，表 3）で用いたハイパーパラメータの値．それぞれの値は検証データで選択した．

	SR	WD	UTR	LM	WR	WR-L	CWR	CWR-L
Twitter(En)	0.2	0.5	0.4	0.4	0.4	0.4	0.4	0.5
+BERT	-	0.3	0.1	0.1	0.2	0.3	0.2	0.2
Twitter(Ja)	0.8	0.5	0.4	0.3	0.4	0.4	0.4	0.4
Weibo(Zh)	0.9	0.3	0.4	0.1	0.2	0.2	0.1	0.4
Rating(En)	0.1	0.4	0.3	0.3	0.3	0.4	0.5	0.5
+BERT	-	0.4	0.1	0.3	0.3	0.4	0.4	0.2
Genre(En)	0.3	0.6	0.7	0.3	0.3	0.3	0.5	0.5
+BERT	-	0.5	0.5	0.4	0.4	0.3	0.5	0.5
Rating(Ja)	0.8	0.3	0.4	0.2	0.3	0.3	0.1	0.4
Genre(Ja)	0.7	0.5	0.5	0.2	0.1	0.2	0.5	0.4
Rating(Zh)	0.5	0.4	0.4	0.2	0.2	0.2	0.7	0.3
Genre(Zh)	0.3	0.3	0.4	0.2	0.2	0.2	0.2	0.2
DeEn	0.5	0.2	0.1	0.1	0.1	0.1	0.2	0.4
EnDe	0.5	0.2	0.2	0.1	0.1	0.2	0.1	0.1
ViEn	0.5	0.2	0.2	0.1	0.2	0.3	0.2	0.5
EnVi	0.5	0.3	0.2	0.1	0.2	0.2	0.2	0.4
ZhEn	0.5	0.2	0.1	0.1	0.1	0.2	0.3	0.1
EnZh	0.4	0.3	0.3	0.2	0.2	0.1	0.4	0.2

Algorithm 1 Algorithm for Building Candidates

```
S ← Empty Dictionary of Set
for Each Sentence in Training Data do
    for Each Substring x ∈ V in Sentence do
        for Each Substring x̃ ∈ V in Sentence do
            if x̃ Partly Overlaps with x then
                ADD x̃ to S_x
            end if
        end for
    end for
end for
```

A　データセット

3.1 節では，文書分類の実験のために 9 つのデータセットを用いた．Twitter(En)[6] と Weibo(Zh)[7] については，配布されているデータセットをそのまま用いた．Twitter(En) のサンプル数は 100,000 件，Weibo(Zh) は 671,052 件である．その他のデータセットについては，以下の通り作成した．

Twitter(Ja)[8]: Twitter API を用いて 352,554 件のツイートを収集し，そのうち感情ラベルが一つのみ付与された 162,184 件 (ポジティブ 10,319 件，ネガティブ 16,035 件，ニュートラル 135,830 件) を実験に利用した．

6） https://www.kaggle.com/c/twitter-sentiment-analysis2
7） https://github.com/wansho/senti-weibo
8） http://www.db.info.gifu-u.ac.jp/data/Data_5d832973308d57446583ed9f

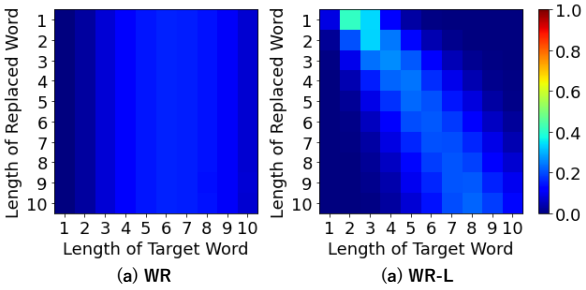

図 4: Amazon データセットでの (a)WR と (b)WR-L による単語置換における，置換対象の単語と置換後の単語の長さの分布．WR-L は WR-L は置換対象の単語の長さに似た長さの単語を頻繁に用いる．

Rating&Genre(En): 配布されている Amazon データセットのうち，サンプル数の量が十分にある 24 の商品ジャンルのレビューから 5,000 件ずつサンプリングした．このとき，レビューの長さは 200 単語以下になるように制約を設けた．単語数はスペース区切りでカウントし，最終的なサンプル数は 120,000 件である．

Rating&Genre(Ja): 配布されている楽天市場のデータセットのうち，サンプル数の量が十分にある 21 の商品ジャンルのレビューから 5,000 件ずつサンプリングした．最大文字数は 100 文字とし，最終的に 525,000 件のサンプルを使用した．

Rating&Genre(Zh): 配布されている JD.com のデータセットのうち，サンプルの量が十分にある 13 の商品ジャンルのレビューから 6,000 件ずつサンプリングした．最大文字数は 100 文字とし，最終的に 390,000 件のサンプルを使用した．

すべてのデータセットは学習，検証，評価データとして 8:1:1 に分割して実験に用いた．また，Rating と Genre タスクはそれぞれ同じデータから作成した．