Vol.2021-NL-250 No.4
2021/9/28

複数の BERT モデルを利用した Data Augmentation

高萩 恭介[1,a)]　新納 浩幸[2,b)]

概要：自然言語処理の分野における簡易な Data Augmentation の手法として，文中の単語をその類似単語に置き換えるというものがある．しかしこの手法は BERT のような事前学習済みモデルを利用する場合には，効果が期待できない．類似単語の知識が BERT に既に組み込まれていると考えられるからである．ここではタスク処理に利用する BERT とは別の BERT の Masked Language Model を利用して類似単語を得ることを提案する．この場合，タスク処理用の BERT に組み込まれていない類似単語の知識を利用できるために，Data Augmentation の効果が期待できる．実験では livedoor ニュースコーパスを利用した文書分類タスクに対して，提案手法による Data Augmentation を試み，提案手法の有効性を示した．

キーワード：Data Augmentation, BERT, MLM, 文書分類

Data Augmentation using multiple BERT models

Takahagi Kyosuke[1,a)]　Shinnou Hiroyuki[2,b)]

1. はじめに

Data Augmentation とは基本的にラベル付きデータに対してラベルはそのままにし，データに小さな変換を施したものを新規のラベル付きデータとして訓練データに加えることで，教師あり学習の性能向上を図る手法である．画像処理の分野では Data Augmentation の効果は高く，教師あり学習を行う際には標準的に利用されている [7]．一方，自然言語処理の分野では取り扱うデータが本質的には離散のデータであるため Data Augmentation のための変換が単純ではなく，大きな成果が出ているとは言いがたい．

自然言語処理の分野における Data Augmentation のための変換は単純ではないが，一つの簡易な変換として文中の単語をその単語の類似単語に置き換えるというものがある [4]．ただし BERT[3] のような事前学習済みモデルを利用する場合には，類似単語の知識は既にモデルの中に組み込まれており，この変換方法では Data Augmentation の効果は出ないと考えられている．そこで本論文ではタスクに利用する BERT とは別種の BERT を用意し，その別種の BERT から類似単語を得る手法を提案する．この場合，タスクに利用する BERT が持たない類似単語の知識を利用することが可能になり，Data Augmentation の効果が出ることが期待できる．実験では livedoor ニュースコーパス[*1]を利用した文書分類タスクに対して，提案手法による Data Augmentation を試み，提案手法の有効性を示した．

2. 関連研究

画像処理の分野では画像を反転させる，切り出すなどといった単純な変換が Data Augmentation として機能し，その効果も高いために，Data Augmentation の利用は，画像処理の際の一般的な処理手順に含まれている．当初はラベル付きデータに対する簡易な小さな変換から始まったが，現在は様々な方向に裾野を広げている．例えばラベル無しデータに対して何らかのラベルを付与して，訓練データを増やす手法も Data Augmentation の一種と見なせる

[1] 茨城大学工学部情報工学科
Department of Computer and Information Sciences, College of Engineering, Ibaraki University
[2] 茨城大学大学院理工学研究科情報科学領域
Graduate School of Science and Engineering, Department of Computer and Information Sciences, Ibaraki University
[a)] 18t4056g@vc.ibaraki.ac.jp
[b)] hiroyuki.shinnou.0828@vc.ibaraki.ac.jp

[*1] https://www.rondhuit.com/download.html#ldcc

ⓒ 2021 Information Processing Society of Japan

ため，従来の半教師あり学習に置き換わっている感もある．また GAN などの生成系の手法も訓練データを追加する形になっており，これらも Data Augmentation の一種と見なすことができる [4].

このように画像処理の分野で活発に研究が行われている Data Augmentation であるが，自然言語処理の分野においては，Data Augmentation 手法の核となる変換手法が単純ではないために，大きな成果は出ていない状況である．ただしいくつかの試みは行われている．

Wei らは EDA (Easy Data Augmentation) という言語モデルや外部データを必要としない簡易な Data Augmentation の手法を提案した [8]．具体的には，訓練データの各文に対して，同義語置き換え，ランダム挿入，ランダム交換あるいはランダム削除の 4 つの操作を複数回ランダムに適用することによって，教師データを拡張する．同じように Kumar らは多様性の高い言い換え表現を得るために，単調劣モジュラ関数の最大化という観点を導入している [6]．また言語 A で記述された文書を言語 B に翻訳し，その翻訳文を更に言語 A に翻訳する，いわゆる逆翻訳を利用して Data Augmentation を行う試みも多い [9][1].

画像処理の分野では Mixup と呼ばれるデータとラベルを各々線形結合したデータを作成する Data Augmentation の手法が提案されている [10]．Gun らはこれを文の埋め込み表現あるいは単語の埋め込み表現を混合する Sentence Mixup や Word Mixup を提案している [5]．Chen らは BERT のある層を混合する TMix に半教師あり学習を併用した MixText を提案している [2].

3. 複数の BERT による Data Augmentation

3.1 別種 BERT による類似単語の生成

BERT モデルは，Masked Language Model(以下「MLM」という) というモデルによっても学習が行われているため，BERT モデル自体に MASK された単語を推定する機構が含まれている．この機構は，BertForMaskedLM という仕組みを用いて BERT モデルから取り出すことができる．

本研究では，まず，この BertForMaskedLM を用いて，学習用データセット中の特定の文章の単語に対応する類似単語を生成する．そして，元の単語を生成した類似単語に置き換えて新しいテキストを生成し，生成したテキストをデータセットに追加するというアプローチで Data Augmentation を行う．具体的な Data Augmentation の方法については，図 1 及び以下に示す.

まず，トークナイザーを使って特定のテキストを単語列に分割する．そして，その単語列を BERT モデルの語彙リスト vocab.txt に基づいた id 列に変換する．その後，MASK したい単語に対応する id を MASK トークンの id に置き換え，置き換え後の id 列をモデルに与える．すると，モデルの出力として id 列に対応した各単語の位置に現れる単語の分布が得られる．その単語分布から，MASK の位置に現れる程度（確率）が大きい単語をいくつか取り出す．そして，取り出した単語の中から最も置換に適した単語を選ぶ．その単語で元のテキストの MASK 位置の単語を置き換えて新しいテキストとし，元のデータセットに追加する．

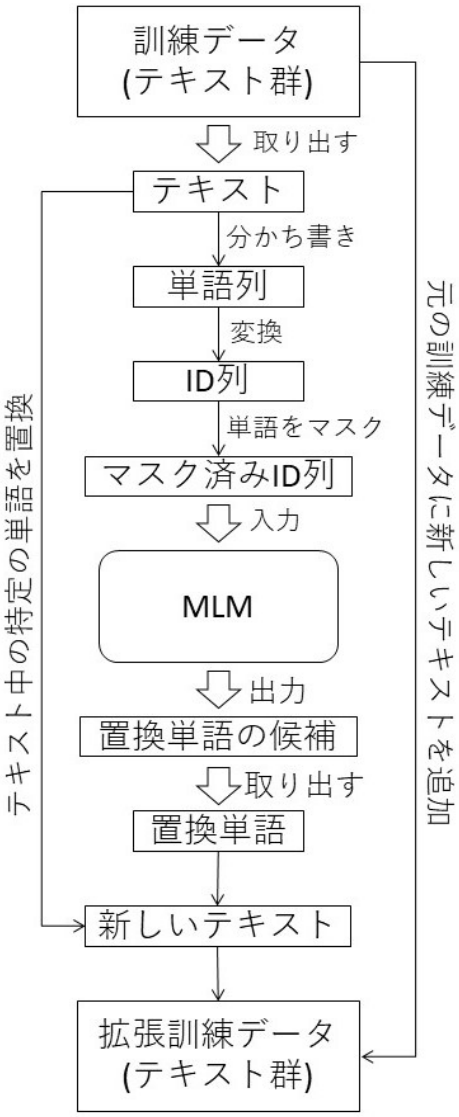

図 1 本研究で提案する Data Augmentation の方法

また，本研究では，Data Augmentation のために，東北大学の乾・鈴木研究室が公開している'bert-base-japanese' という BERT モデル[*2] (以下「東北大版 BERT」という) とストックマーク株式会社が公開している BERT モデル[*3] (以下「ストックマーク版 BERT」という) の，2 つの異なる BERT モデルの MLM を利用する．東北大版 BERT は，事前学習用データとして日本語の Wikipedia を用いてい

[*2] https://github.com/cl-tohoku/bert-japanese
[*3] https://qiita.com/mkt3/items/3c1278339ff1bcc0187f

る．また，トークナイザーには MeCab(Unidic2.1.2 辞書) と WordPiece を使っている．ストックマーク版 BERT はビジネスニュース向けの BERT モデルで，事前学習用データとして日本語ビジネスニュース記事を利用している．また，トークナイザーには MeCab + NEologd を利用している．

そして，ベースラインとなる学習用データセットにこれら 2 つの BERT モデルの MLM を使った Data Augmentation を適用し，それぞれに対応した拡張データセットを作成する．

3.2 文書分類に対する変換文の生成

Data Augmentation によって新しいテキストを生成するには，元のデータセットの中から新しいテキストの元となるテキストとそのテキスト中の置き換え元の単語を適切な方法で選択する必要がある．本研究では，データセットのテキスト群に含まれる全ての単語の中から TF-IDF の値が高い単語を選択後，その単語が置き換え元の単語として適切であるかどうかを判定する．そして，適切であると判定された場合はその単語を MASK し，類似単語へ置き換えることにする．

まず，Python の機械学習ライブラリである scikit-learn の TfidfVectorizer を使って，データセットのテキスト群を TF-IDF ベクトルに変換する．この時，TfidfVectorizer のトークナイザーには，Data Augmenation に用いる BERT モデルのトークナイザーを指定する．次に，この TF-IDF ベクトルを元にして，データセットのテキスト群に含まれる全ての単語を TF-IDF 値の大きい順にソートする．そして，TF-IDF 値が大きい単語から順に，類似単語への変換処理を行う．

変換処理では，まず，選択中の単語が以下の条件を満たすかどうかを判定する．

- 単語が名詞である
- 単語が BERT モデルの語彙リスト vocab.txt に存在する

これらの条件を全て満たす時，選択中の単語があるテキストの選択中の単語を MASK して，そのテキストを BERT の MLM に渡す．その後，MLM の出力から MASK の位置に現れる程度（確率）が大きい単語の上位 5 つを取り出す．取り出した 5 つの単語について，上位の単語から順に以下の条件を満たすかどうかを判定していく．

- 単語が名詞である
- 単語が BERT モデルの語彙リスト vocab.txt に存在する
- 単語が MASK した単語と一致していない．
- 単語が MASK した単語と部分的に一致していない．

これらの条件を満たす最上位の単語で，置き換え元の単語を置換し，置換後の新しいテキストと置換前のテキストのラベルのセットを元のデータセットに追加する．ここまでの変換処理を，拡張中のデータセットのデータ数が指定した数に到達するまで繰り返し行う．

4. 実験

4.1 実験設定

本実験では，タスクとして文書分類を用いる．また，文書分類モデルの学習・評価に用いるデータセットには，livedoor ニュースコーパス[*4]を利用することとする．livedoor ニュースコーパスは，NHN Japan 株式会社が運営する「livedoor ニュース」の中からニュース記事を収集し，可能な限り HTML タグを取り除いて作成したものである．このコーパスには，9 つのカテゴリのニュース記事が格納されている．ここでは，各カテゴリに対してラベルを割当てた後，コーパスから各ラベルのニュース記事を同じ数だけランダムに取り出し，訓練データ，検証データ，テストデータの 3 つデータセットを作成する．このときのニュース記事のカテゴリとそれに対応するラベル，各データセットに含まれる各カテゴリの記事数は，表 1 の通りである．

表 1 ニュース記事のカテゴリとそれに対応するラベル，各データセットに含まれる各カテゴリの記事数

ラベル	カテゴリ	train	val	test
0	独女通信	30	30	30
1	IT ライフハック	30	30	30
2	家電チャンネル	30	30	30
3	livedoor HOMME	30	30	30
4	MOVIE ENTER	30	30	30
5	Peachy	30	30	30
6	エスマックス	30	30	30
7	Sports Watch	30	30	30
8	トピックニュース	30	30	30
	合計	270	270	270

ここで作成した訓練データが baseline となる．この訓練データに対して，3 章に記した Data Augmentation の手法を適用し，拡張訓練データを作成する．作成するものは，東北大版 BERT の MLM を利用した拡張訓練データ (以下「拡張訓練データ (東北大)」という) と，ストックマーク版 BERT の MLM を利用した拡張訓練データ (以下「拡張訓練データ (ストックマーク)」という) の 2 種類である．データ数は 2 種類とも 370 個であり，その内訳は元のデータが 270 個で Data Augmentation で新たに生成したデータが 100 個である．

そして本実験では，ここまでに作成した 3 つの訓練データを評価するために，BERT を利用した文書分類プログラムを実装する．文書分類プログラムについても，ネットワークのモデルに東北大版 BERT を利用したものとストックマーク版 BERT を利用したものの 2 種類を実装する．

[*4] https://www.rondhuit.com/download.html#ldcc

文書分類の基本的な流れとしては，各訓練データを学習することで文書分類モデルを構築し，そのモデルを利用してテストデータの識別を行う．ここで使用するモデルは，BERT の出力に識別の層である全結合層をつないだものである．具体的なモデルの構造としては，512 単語以下の ID 列が BERT に入力され，入力列の先頭 token の [CLS] に対する BERT の出力である 768 次元ベクトルが，全結合層で 9 次元ベクトルに変換される．モデルの最終的な出力である 9 次元ベクトルの各要素はデータセットの各ラベルに対応していて，9 次元ベクトルの最大要素のインデックスが予測結果のラベルとなっている．

また，今回の実験で用いる学習プログラムでは，最適化アルゴリズムに SGD（確率的勾配降下法）を設定し，損失関数にクロスエントロピーを設定している．学習プログラムのその他の設定としては，バッチサイズとエポック数がある．バッチサイズは 2 にしている．エポック数はアーリーストッピングを用いて決定している．具体的には，まず，学習が 1 エポック終了するごとにその段階で構築されたモデルを検証データを使って評価し，検証データに対する正解率を出す．このとき，正解率が 5 エポック連続でそのときまでの最高値を下回った場合は過学習が起こっていると判断し，それまでの最高値となる正解率を出したモデルを最終的に構築したモデルとする．アーリーストッピングによる学習の中断がなかった場合は，30 エポックで学習を打ち切る．

それから，今回用いた学習プログラムでは，モデルの全結合層の重みの初期値が毎回異なるため，学習によって構築されるモデルも毎回異なる．そのため，同じ学習プログラム・訓練データを使ったモデルの構築を 5 回繰り返し，できた 5 つのモデルの中から検証データでの正解率が一番高かったものを選んでテストデータの評価を行う．そして，得られたテストデータの正解率を実験結果として用いる．

4.2 実験結果

まず，ネットワークのモデルに東北大版 BERT を利用した学習プログラムを利用して，baseline の訓練データ，拡張訓練データ (東北大)，拡張訓練データ (ストックマーク) の 3 つの訓練データからそれぞれに対応する文書分類モデルを構築した．その後，構築した各モデルを使ってテストデータの識別を行い，各モデルについての正解率を出した．このときの結果を表 2 及び図 2 に示す．

表から分かるように，拡張訓練データ (ストックマーク) の正解率は，baseline の正解率を上回った．また，拡張訓練データ (東北大) の正解率は，baseline の正解率を下回った．

表 2 学習に東北大版 BERT を用いたモデルの正解率

訓練データ	正解率
baseline	0.8629
拡張訓練データ (東北大)	0.8481
拡張訓練データ (ストックマーク)	0.8703

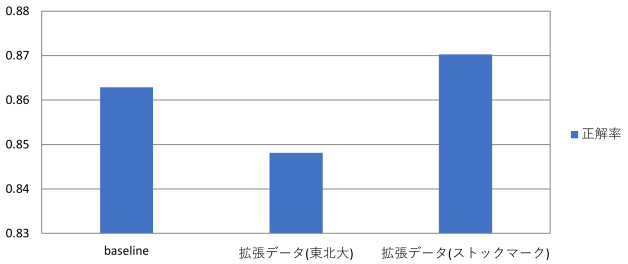

図 2 学習に東北大版 BERT を用いたモデルの正解率

次に，ネットワークのモデルにストックマーク版 BERT を利用した学習プログラムを利用して，baseline の訓練データ，拡張訓練データ (東北大)，拡張訓練データ (ストックマーク) の 3 つの訓練データからそれぞれに対応する文書分類モデルを構築した．その後，構築した各モデルを使ってテストデータの識別を行い，各モデルについての正解率を出した．このときの結果を表 3 及び図 3 に示す．

表から分かるように，拡張訓練データ (東北大) の正解率，拡張訓練データ (ストックマーク) の正解率は，共に baseline の正解率を上回った．また，拡張訓練データ (東北大) の正解率と拡張訓練データ (ストックマーク) の正解率を比較すると，前者のほうが高くなった．

表 3 学習にストックマーク版 BERT を用いたモデルの正解率

訓練データ	正解率
baseline	0.8333
拡張訓練データ (東北大)	0.8629
拡張訓練データ (ストックマーク)	0.8592

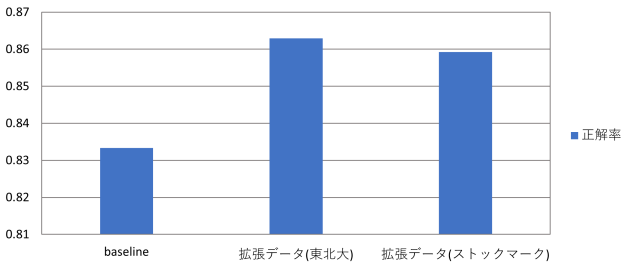

図 3 学習にストックマーク版 BERT を用いたモデルの正解率

ⓒ 2021 Information Processing Society of Japan

5. 考察

5.1 2つの BERT の類似単語の重なり

提案手法が効果を発揮するのは，ある単語と類似する単語が 2 つの BERT で異なるからである．ここでは，実験で利用した訓練データのテキスト群に含まれる単語に対して，東北大版 BERT とストックマーク版 BERT によって得られる類似単語がどの程度異なるかを，その重なり具合から調べる．この調査のために以下の実験を行う．

まず，本実験で利用した訓練データの各テキストを分かち書きし，単語に分割する．分かち書きには東北大版 BERT の tokenizer を利用する．次に，分かち書きによって得られたすべての単語の中から，東北大版 BERT とストックマーク版 BERT の両方の語彙リストに含まれる名詞を TF-IDF 値が高い順に 100 個選ぶ．このとき選んだ 100 個の単語一つ一つに対して，3 章で記した方法を用いて 2 つの BERT の類似単語をそれぞれ上位 5 つずつ取得し，その 5 つ同士を比較して 5 個中何個一致するかを調べる．そして，選んだ単語 100 個のうち，0 から 5 までの各一致数に該当する数を求める．また，TF-IDF 値の高い順に単語を選んだ効果を調べるために，単語の選び方のみをランダムに変更して同様に実験を行う．

ここからは，上記の実験を行った結果について示す．TF-IDF 値が高い順に選んだ単語 100 個とランダムに選んだ単語 100 個について，0 から 5 までの各一致数にそれぞれ何個該当したかを表 4 及び図 4 に示した．表 4 の結果を元に計算を行ったところ，TF-IDF 値の高い順に選んだ単語 100 個における 2 つの BERT の類似単語は平均で 5 個中 1.64 個，つまり 32.8 %重なり，ランダムに選んだ単語 100 個における 2 つの BERT の類似単語は平均で 5 個中 0.53 個，つまり 10.6 %重なることが分かった．

表 4 0 から 5 までの各一致数にそれぞれ該当する単語数

	0	1	2	3	4	5	合計
TF-IDF	13	33	34	17	3	0	100
ランダム	68	19	7	4	2	0	100

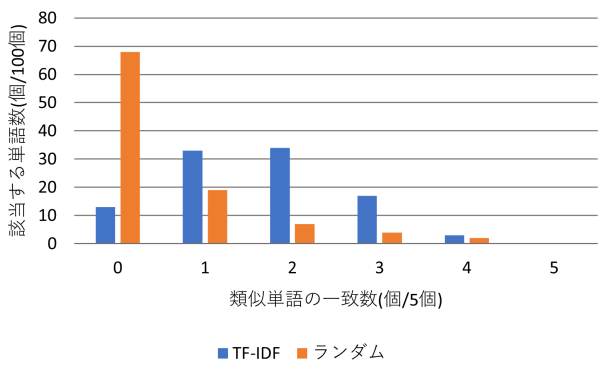

図 4 0 から 5 までの各一致数にそれぞれ該当する単語数

ここで，単語を TF-IDF の高い順に選んだ場合とランダムに選んだ場合の結果を比較すると，ランダムに選んだ場合の類似単語の重なり具合の方が悪いことが分かる．このような結果になったのは，ランダムに選んだ単語に対してストックマーク版 BERT の類似単語取得を行った際に，類似単語として「は」,「を」,「の」,「、」,「。」などを多く取得してしまったことが原因と考えられる．これらの単語は明らかに今回選択した単語に対する類似単語ではなかった．そこで今回は，単語を TF-IDF の高い順に選んだ場合の結果のみについて考察を行うこととする．また，訓練データのテキスト群の分かち書きに使うトークナイザーを東北大版 BERT からストックマーク版 BERT のトークナイザーに変更して再度実験を行ったが，変更前の実験と同じような結果が得られた．

ここまでの結果より，今回の訓練データのテキスト中に含まれる TF-IDF 値が高い単語に対する類似単語は，東北大版 BERT とストックマーク版 BERT で 32.8 %程重なっていることが分かった．ここでの重なった類似単語の知識は，東北大版 BERT とストックマーク版 BERT の両方に含まれると考えられる．一方で，それ以外の重なっていない類似単語の知識の中には，それぞれの BERT に固有の知識が含まれていると考えることができる．今回実験で試した Data Augmentation では，この類似単語に含まれる片方の BERT に固有の知識が訓練データの知識とネットワークモデルに用いたもう片方の BERT の知識に加わったことで，文書分類タスクの精度が向上したと考えられる．

また今回は，東北大版 BERT とストックマーク版 BERT を利用したが，これらの BERT 以外にも様々な BERT が存在する．そこで，BERT 同士の知識があまり重ならないような 2 種類の BERT のペアを選択して，今回と同じように Data Augmentation を行えば，今回以上の精度向上が期待できる．また，Data augmentation に使う BERT の種類を 1 種類ではなく多種類に増やすことによっても，精度の向上が期待できる．

5.2 訓練データの量の影響

提案手法のような簡易な Data Augmentation は，元の訓練データが小さい場合に効果があり，大きい場合には効果がないと考えられている [8]．ここでは，この点を確認するために，4 章で行った実験を訓練データのサイズを 2 倍にして同様に行う．

4 章の実験では，訓練データに baseline の訓練データ，拡張訓練データ (東北大)，拡張訓練データ (ストックマーク) の 3 つ，ネットワークのモデルに東北大版 BERT とストックマーク版 BERT の 2 つを利用し，計 6 パターンのモデルを作成した．しかし，ここでの実験では，訓練データに baseline の訓練データ，拡張訓練データ (ストックマーク) の 2 つ，ネットワークのモデルには東北大版 BERT を

利用し，計 2 パターンのモデルを作成する．

まず，4 章の実験と同様に livedoor ニュースコーパスを用いて，baseline の訓練データ，検証データ，テストデータを作成する．ここでは訓練データのサイズを 2 倍とするため，baseline の訓練データのデータ数は 540 個とし，使用するデータは 4 章の実験で使った baseline の訓練データとは全く異なるものにした．また，検証データとテストデータについては 4 章の実験で使ったものと全く同じものを使用する．

次に，baseline の訓練データに対して，3 章に記した Data Augmentation の手法を適用して拡張訓練データを作成する．4 章の実験では拡張訓練データ (東北大) と拡張訓練データ (ストックマーク) の 2 種類を作成したが，ここでは拡張訓練データ (ストックマーク) のみを作成する．データ数は 740 個で，その内訳は baseline の訓練データが 540 個，Data Augmentation で新たに生成したデータが 200 個である．

そして，作成した 2 種類の訓練データを評価するために，BERT を利用した文書分類プログラムを実装する．4 章の実験ではネットワークのモデルに東北大版 BERT を利用したものとストックマーク版 BERT を利用したものの 2 種類を実装したが，ここではその時実装した 2 つのうち，東北大版 BERT を利用したもののみを用いる．また，文書分類プログラムの各種設定や評価用モデルの選出方法については 4 章の実験と全く同じにしている．

ここからは上記の実験を行った結果について示す．ネットワークのモデルに東北大版 BERT を利用した学習プログラムを利用して，baseline の訓練データと拡張訓練データ (ストックマーク) のそれぞれに対応する文書分類モデルを構築した．その後，構築した 2 つのモデルを使ってテストデータの識別を行い，それぞれについて正解率を出した．このときの結果を表 5 及び図 5 に示す．

表から分かるように，正解率は baseline と拡張訓練データ (ストックマーク) で変わらないという結果になった．

この結果から，本論文で提案した Data Augmentation の手法は，訓練データのサイズが小さい場合には効果を発揮するが，サイズが大きくなると効果がなくなることを確認できた．つまりこの手法は，適用するタスクにおいて訓練データの数が十分ではない場合には適しているが，十分な数の訓練データがあるときには実用性は期待できない．

今後はデータの量が中規模であっても効果が出るような Data Augmentation の手法を考案したい．そこでは複数の BERT を利用するというアイデアが使えると考えている．

表 5 学習に東北大版 BERT を用いたモデルの正解率 (訓練データ 2 倍)

訓練データ	正解率
baseline	0.8888
拡張訓練データ (ストックマーク)	0.8888

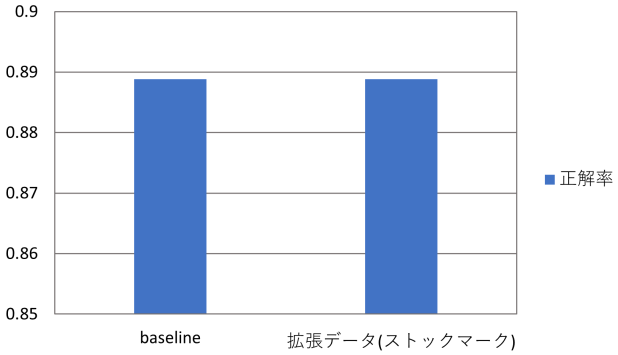

図 5 学習に東北大版 BERT を用いたモデルの正解率 (訓練データ 2 倍)

6. おわりに

本論文では 2 つの BERT を用いた Data Augumentation の手法を提案した．具体的には，タスクに用いる BERT とは別種の BERT の MLM を用いて，訓練データ中の名詞を類似の名詞に置き換えることで，訓練データを増やす．タスクに用いる BERT が持っていない知識を利用できる形になるために，Data Augumentation の手法として成り立つ．実験では livedoor ニュースコーパスをデータとした文書分類をタスクとし，東北大版 BERT とストックマーク版 BERT の 2 つの BERT を用いて提案する Data Augumentation の手法の有効性を示した．今後は複数の BERT を利用した別種の Data Augmentation の手法を試みたい．

謝辞

本研究は JSPS 科研費 JP19K12093 および 2021 年度国立情報学研究所公募型共同研究（2021-FC05）の助成を受けています．

参考文献

[1] Chen, J., Wu, Y. and Yang, D.: Semi-supervised models via data augmentationfor classifying interactive affective responses, *arXiv preprint arXiv:2004.10972* (2020).

[2] Chen, J., Yang, Z. and Yang, D.: MixText: Linguistically-Informed Interpolation of Hidden Space for Semi-Supervised Text Classification, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, Association for Computational Linguistics, pp. 2147–2157 (online), DOI: 10.18653/v1/2020.acl-main.194 (2020).

ⓒ 2021 Information Processing Society of Japan

[3] Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, *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)*, Minneapolis, Minnesota, Association for Computational Linguistics, pp. 4171–4186 (online), DOI: 10.18653/v1/N19-1423 (2019).

[4] Feng, S. Y., Gangal, V., Wei, J., Chandar, S., Vosoughi, S., Mitamura, T. and Hovy, E.: A Survey of Data Augmentation Approaches for NLP, *arXiv preprint arXiv:2105.03075* (2021).

[5] Guo, H., Mao, Y. and Zhang, R.: Augmenting data with mixup for sentence classification: An empirical study, *arXiv preprint arXiv:1905.08941* (2019).

[6] Kumar, A., Bhattamishra, S., Bhandari, M. and Talukdar, P.: Submodular Optimization-based Diverse Paraphrasing and its Effectiveness in Data Augmentation, *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)*, Minneapolis, Minnesota, Association for Computational Linguistics, pp. 3609–3619 (online), DOI: 10.18653/v1/N19-1363 (2019).

[7] Shorten, C. and Khoshgoftaar, T. M.: A survey on image data augmentation for deep learning, *Journal of Big Data*, Vol. 6, No. 1, pp. 1–48 (2019).

[8] Wei, J. and Zou, K.: Eda: Easy data augmentation techniques for boosting performance on text classification tasks, *arXiv preprint arXiv:1901.11196* (2019).

[9] Xia, M., Kong, X., Anastasopoulos, A. and Neubig, G.: Generalized Data Augmentation for Low-Resource Translation, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, Association for Computational Linguistics, pp. 5786–5796 (online), DOI: 10.18653/v1/P19-1579 (2019).

[10] Zhang, H., Cisse, M., Dauphin, Y. and Lopez-Paz, D.: mixup: Beyond empirical risk minimization. ICLR 2018, *arXiv preprint arXiv:1710.09412* (2017).

ⓒ 2021 Information Processing Society of Japan