Vol.2022-NL-253 No.1
2022/9/29

平仮名 BERT による平仮名文の分割

井筒 順[1,a)]　古宮 嘉那子[2,b)]　新納 浩幸[3,c)]

概要：日本語を形態素に解析するために MeCab や Chasen 等の形態素解析システムが存在している。現在存在している日本語の形態素解析システムの精度は非常に高いが、これらのシステムは漢字仮名混じりの文を対象にしているため、平仮名で書かれた文を形態素に分割することは難しい。これは文がほとんど全て平仮名で書かれていることにより、形態素として分割する場所の特定が難しくなるからである。本研究では unigram BERT と bigram BERT の 2 種類の BERT による平仮名文の単語分割モデルを作成した。BERT モデルの作成に際し、事前学習用データとして Wikipeda のデータを用い、単語分割のためのファインチューニングのデータとして BCCWJ のコアデータを利用した。さらに、作成した 2 種類の BERT による平仮名文の単語分割における精度と比較するため、Kytea を用いた平仮名文の単語分割モデルを作成した。BCCWJ のコアデータを用い 5 分割交差検証を行ったところ、unigram BERT 単語分割システムでは 97.67%の精度を、bigram BERT 単語分割システムでは 96.44%の精度を得た。

1. はじめに

日本語を形態素に解析するために MeCab[*1] や Chasen[*2] 等の形態素解析システムが存在している。現在存在している日本語の形態素解析システムの精度は非常に高いが、これらのシステムは漢字仮名混じりの文を対象にしているため、これらを用いてほとんど全てが平仮名で書かれた文[*3] を形態素に分割することは難しい。

本稿では unigram BERT と bigram BERT の 2 種類の BERT[4] (Bidirectional Encoder Representations from Transformers) による平仮名文分割モデルを作成した。BERT モデルの作成に際し、事前学習用データとして、MeCab を用いて分かち書きを行った Wikipeda のデータの読みの部分を平仮名に変換したものを利用した。また、平仮名分割モデルのファインチューニングのデータとして国立国語研究所の『現代日本語書き言葉均衡コーパス』[*4](以下 BCCWJ[9] と記す) のコアデータを利用した。ファインチューニング用のデータも事前学習用データと同様、BCCWJ のコアデータにおける読みの部分を平仮名に変換したものを利用している。さらに、作成した 2 種類の BERT による平仮名文分割モデルの精度と比較するため、KyTea[*5] を用いた平仮名文の分割モデルを作成した。

BCCWJ のコアデータを用い 5 分割交差検証を行った実験において、unigram BERT では 97.67%の精度を、bigram BERT では 96.44%の精度を得た。

2. 関連研究

平仮名文の単語分割および形態素解析の論文には以下がある。まず、工藤ら [3] は，平仮名交じり文が生成される過程を生成モデルを用いてモデル化した。そして、そのパラメータを大規模 Web コーパスと EM アルゴリズムで推定することで、平仮名交じり文の解析精度を向上させる手法を提案している。また、林ら [5] は平仮名語の単語を辞書に追加することで形態素解析の精度が向上することを報告している。井筒ら [1] は MeCab の ipadic 辞書を平仮名に変換し、平仮名のみで構成されたコーパスを用いることで平仮名のみの文での形態素解析を行っている。さらに、井筒ら [2] は Bi-LSTM CRF モデルを用いた平仮名文の形態素解析を行い、複数のジャンルの文に対して複数にわたって学習とファインチューニングを行うことで形態

1 茨城大学大学院理工学研究科情報工学専攻
Major in Computer and Information Sciences, Graduate School of Science and Engineering, Ibaraki University
2 東京農業工業大学大学院工学研究院先端情報科学部門
Institute of Engineering, Division of Advanced Information Technology & Computer Science, Tokyo University of Agriculture and Technology
3 茨城大学大学院理工学研究科情報工学領域
Graduate School of Science and Engineering, Department of Computer and Information Sciences, Ibaraki University
a) 21nm707h@vc.ibaraki.ac.jp
b) kkomiya@go.tuat.ac.jp
c) hiroyuki.shinnou.0828@vc.ibaraki.ac.jp

*1 https://taku910.github.io/mecab/
*2 https://chasen-legacy.osdn.jp
*3 数字や記号は含まれている。
*4 https://pj.ninjal.ac.jp/corpus_center/bccwj/
*5 http://www.phontron.com/kytea/

素解析の精度にどのように変化を与えられるかを報告している。また、森山ら [7] は Recurrent Neural Language Model（RNNLM）を用いたべた書きかな文の形態素解析を行ない、その精度が単語分割と単語素性の全てを正解とする最も厳しい基準において従来手法を有意に上回ることを報告している。さらに、森山ら [8] は Recurrent Neural Network とロジスティック回帰を用いた平仮名文の逐次的な形態素解析手法を提案し、平仮名文における形態素解析の精度向上とシステムの高速化を報告している。

また、本研究では平仮名に特化した平仮名 BERT を作成し、利用することで、平仮名の文の単語分割モデルを作成している。分野に特化した BERT の作成として代表的なものには鈴木ら [6] がある。これは、金融文書を用い金融に関する文に特化した BERT を作成したことを報告する論文である。また、汎用言語コーパスを用いて事前学習を行った BERT モデルに対して、金融コーパスを用いてファインチューニングを行うことが有効であるかの検証を行なっている。

3. 提案手法

BERT は Transformer がベースとなっている自然言語処理モデルである。BERT は Attention 機構を用い、トークンを処理する際に全てのトークンを参照している点に特徴がある。BERT の構造を著した図を**図 1** に示す。

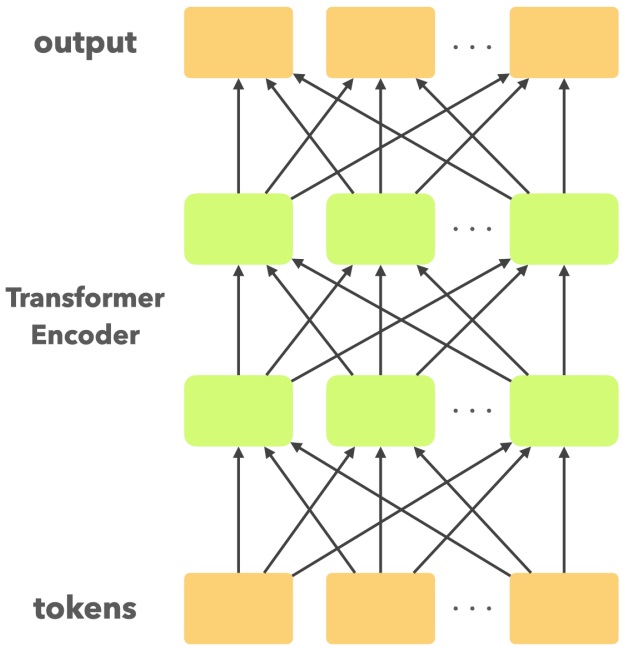

図 1 BERT の構造

本稿では平仮名文に特化した 2 種類の平仮名 BERT モデルを生成し、それぞれを利用して平仮名文の単語分割システムを作成した。平仮名 BERT モデルのうち、1 つ目のモデルは unigram BERT である。これは平仮名の文字 unigram で構成された文集合を事前学習に利用して生成した BERT モデルである。2 つ目のモデルは bigram BERT モデルである。これは平仮名の文字 bigram で構成された文集合を事前学習に利用して生成した BERT モデルである。我々はこのように unigram BERT と bigram BERT を作成し、これらを平仮名文の単語分割のデータを使ってファインチューニングすることで、平仮名の文の単語分割システムを作成した。これら 2 つの平仮名文の単語分割システムの精度を比較する。さらに、KyTea による平仮名文の単語分割のモデルを作成し、unigram BERT と bigram BERT を利用した平仮名文の単語分割システムと比較する。

3.1 unigram BERT 単語分割システム

unigram BERT は、事前学習用データとして平仮名の文字 unigram で構成された文を利用した BERT モデルである。Wikipedia の漢字仮名交じり文を平仮名に変換し、さらに文字 unigram に分かち書きしたデータを事前学習用のデータとして使用した。Wikipedia には平仮名のみのデータが存在しないので、MeCab の解析結果における読みのデータを疑似的な正解として利用した。

また、この unigram BERT に対して、平仮名文の単語分割の情報が付与されたデータを使ってファインチューニングすることで、平仮名文の単語分割システムを作成した。これを以降、unigram BERT 単語分割システムと呼ぶ。ファインチューニングのデータには、実験によって BCCWJ または Wikipedia のいずれかのデータを利用している。

3.2 bigram BERT 単語分割システム

bigram BERT は、事前学習用データとして、平仮名の文字 bigram で構成された文を利用した BERT モデルである。Wikipedia の漢字仮名交じり文を平仮名に変換し、さらに文字 bigram に分かち書きしたデータを事前学習用のデータとして使用した。bigram BERT 単語分割システムの事前学習に使用する Wikipedia は、unigram BERT 単語分割システムの事前学習に使用した Wikipedia と同様に平仮名のみのデータが存在しないので、MeCab の解析結果における読みのデータを疑似的な正解として利用している。

また、この bigram BERT に対して、平仮名文の単語分割の情報が付与されたデータを使用してファインチューニングをすることで、平仮名文の単語分割システムを作成した。これを以降、bigram BERT 単語分割システムと呼ぶ。ファインチューニングのデータには、実験によって BCCWJ または Wikipedia のいずれかのデータを利用して

いる。

3.3 平仮名 KyTea 単語分割システム

京都大学森研究室の開発したテキスト解析ツールキット KyTea を用いて平仮名文の単語分割システムを作成した。KyTea は単語分割および読み推定の機能を持つシステムである。部分的アノテーションから学習をすることが可能であり、点予測を利用して文の解析を行う。KyTea を利用して作成した平仮名文の単語分割システムを以降、平仮名 KyTea 単語分割システムと呼ぶ。学習データは、実験によって BCCWJ または Wikipedia のデータを利用している。

4. データ

4.1 Wikipedia による事前学習用データ

2 種類の平仮名 BERT（unigram BERT と bigram BERT）を作成するための事前学習用のデータとして、Wikipedia を利用した。このデータは以下の Web サイト

https://dumps.wikimedia.org/jawiki/latest/

において公開されているデータの

jawiki-latest-pages-articles.xml.bz2

を展開し使用した。

Wikipedia による平仮名の分かち書きデータの作成方法について論述する。Wikipedia は漢字仮名混じり文であるため、平仮名文に変換するには MeCab を用いる。まず、Wikipedia の漢字仮名交じり文を MeCab を利用して形態素解析し、形態素解析結果における読み部分を利用することで平仮名のみで構成された文を得る。MeCab の辞書には Unidic を利用した。このように MeCab の読みデータから作成しているため、Wikipedia を用いて作った平仮名文は、正確な平仮名文ではなく、疑似的な平仮名文である。次に、平仮名のみで構成された文を文字 unigram の形と文字 bigram の形に変換した。文字 unigram の形に変換したデータは unigram BERT の事前学習用データとして利用し、文字 biram の形に変換したデータは bigram BERT の事前学習用データとして利用する。最後に、文の行頭と行末に対してそれぞれ [CLS] タグと [SEP] タグを付与した。

unigram BERT と bigram BERT の事前学習用データ例を**表 1** に示す。

表 1 事前学習用データ例

元データ	今日は寒い。
unigram 用	[CLS] き ょ う は さ む い 。 [SEP]
bigram 用	[CLS] きょ ょう うは はさ さむ むい い。 [SEP]

上記の操作により 300 万文の Wikipedia による事前学習用データを得た。作成したデータの中身に関しては文字 bigram で表現されているのか文字 unigram で表現されているのかを除けば同一のデータを利用している。

4.2 Wikipedia による平仮名文の単語分かち書きデータ

unigram BERT と bigram BERT におけるファインチューニングに利用するデータとして、Wikipeda による平仮名文の単語分かち書きデータを作成した。本データの作成方法については 4.1 を参照されたい。ただし、4.1 とは異なり [CLS] タグと [SEP] タグは付与していない。また、4.1 とは異なり 0 と 1 で構成されたタグ情報も作成した。Wikipedia の漢字仮名交じり文を MeCab を利用して形態素解析し、各単語における読みの部分について先頭を 1、それ以降を 0 とすることで 0 と 1 で構成されたタグデータを得た。

Wikipedia による平仮名文の単語分かち書きデータ例を**表 2** に示す

表 2 Wikipedia による平仮名文の単語分かち書きデータタ例

元データ	今日は寒い。
unigram 用	き ょ う は さ む い 。
bigram 用	きょ ょう うは はさ さむ むい い。
タグデータ	100110011

上記の操作により 100 万文の Wikipedia による平仮名文の単語分かち書きデータを得た。作成したデータの中身に関しては文字 unigram で表現されているのか文字 bigram で表現されているのかを除けば同一のデータを利用している。

4.3 BCCWJ による平仮名文の単語分かち書きデータ

BCCWJ のコアデータは人手により形態素に分けられたデータである。Wikipedia による事前学習用データと Wikipedia による平仮名文の単語分かち書きデータでは正確ではない擬似的な平仮名文を作成したが、BCCWJ のコアデータを利用することにより正確な平仮名文を作成することが可能となる。BCCWJ コアデータを平仮名の分かち書きに変換したデータを平仮名文の単語分割システムのファインチューニングおよびテストデータとして利用する。実験による使用データの違いについての詳細は 5 節で述べる。

次に本データの作成方法について論述する。まず BCCWJ コアデータの読みの部分を利用しほぼ平仮名で構成された文に変換した。そして、ほぼ平仮名で構成された文を文字 unigram の形と文字 bigram の形に変換した。文字 unigram の形に変換したデータは unigram BERT のファインチューニングデータとして利用し、文字 biram の形に変換したデータは bigram BERT のファインチューニングデータとして利用する。

また BCCWJ コアデータの読みの部分を利用することで 0 と 1 のみで表現されたタグデータを作成した。形態素に分割された読み部分の先頭を 1 に、それ以降を 0 とすることで 0 と 1 で構成されたタグデータを得た。

BCCWJ による平仮名文の単語分かち書きデータ例を

表 3 に示す。

表 3 BCCWJ による平仮名文の単語分かち書きデータ

元データ	今日は寒い。
unigram 用	き ょ う は さ む い 。
bigram 用	きょ ょう うは はさ さむ むい い。
タグデータ	100110011

上記の操作により 40928 文の BCCWJ による平仮名の分かち書きデータを得た。

4.4 平仮名 KyTea 単語分割システムのデータ

平仮名文を単語する分割ができる KyTea モデルの作成を行うために、学習データとして BCCWJ のコアデータによる平仮名の分かち書きデータを利用した。本データの読みの部分を利用しほぼ平仮名のみで構成された分かち書きデータを得た。

平仮名 KyTea 単語分割システム作成に使用したデータ例を**表 4** に示す。

表 4 平仮名 KyTea 単語分割システム作成に使用したデータ

元データ	今日は寒い。
学習データ	きょう は さむい 。

4.5 平仮名 BERT の語彙

unigram BERT 作成において使用した語彙の総数は 300 である。語彙には平仮名、カタカナ、アルファベット、数字、複数の記号が含まれている。また、bigram BERT 作成において使用した語彙の総数は 80956 である。語彙には平仮名、カタカナ、アルファベット、数字、複数の記号の任意の 2 種類の組み合わせが含まれている。

5. 実験

作成した 2 種類の BERT における平仮名文の単語分割の精度がファインチューニング時のデータ量とデータの種類によりどのように変化するのかを検証するために 2 つの実験を行った。

5.1 実験 1：BCCWJ によるファインチューニングの実験

1 つ目の実験は、BCCWJ によるファインチューニングの実験である。この実験では、正確な平仮名文の単語分割情報のデータを利用して、unigram BERT 単語分割システムと bigram BERT 単語分割システムの精度をベースラインとなる平仮名 KyTea 単語分割システムの精度と比較する。この実験では平仮名 BERT 作成のための事前学習用データとして 300 万文の Wikipedia による事前学習用データを利用した。そしてファイチューニングのデータおよびテストデータとして 40928 文の BCCWJ による平仮名文の単語分かち書きデータを利用し、5 分割交差検定を行った。BCCWJ による平仮名文の単語分かち書きデータの内 5 分の 3 をファインチューニングのデータとして利用し、5 分の 1 を検証用データとして利用し、5 分の 1 のデータをテストデータとして利用している。また平仮名 KyTea 単語分割システムについても、、unigram BERT 単語分割システムと bigram BERT 単語分割システムのファインチューニング時に利用したものと同様に BCCWJ による平仮名文の単語分かち書きデータを利用し、5 分割交差検定を行った。 利用した学習データはの BCCWJ による平仮名文の単語分かち書きデータの 5 分の 4 である。この実験を可視化した図を**図 2** に示す。

次に、本実験において BERT の事前学習で使用したパラメータを**表 5** に、ファインチューニングで使用したパラメータを**表 6** に示す。

表 5 事前学習におけるパラメータ

レイヤー数	12
隠れ層	120
学習率	1e-4
バッチサイズ	8
ステップ数	1000000

表 6 ファインチューニングにおけるパラメータ

ラベル数	12
学習率	1e-5
エポック数	24

5.2 実験 2：Wikipedia によるファインチューニングの実験

2 つ目の実験は、Wikipedia によるファインチューニングの実験である。この実験では、疑似データである Wikipedia の単語分割情報を大量に利用した場合の三つの作成した単語分割システムの精度を見る。この実験では BERT 作成時の事前学習用データとして 300 万文の Wikipedia による事前学習用データを利用し、ファイチューニングのデータとして 100 万文の Wikipedia による平仮名文の単語分かち書きデータを利用した。事前学習用のデータとファインチューニング用のデータの重複はない。一方で、実験 1 における事前学習データと実験 2 における事前学習データは同一のものを利用している。また KyTea の学習データとして、unigram BERT および bigram BERT のファインチューニングで利用したものと同じ 100 万文の Wikipedia による平仮名文の単語分かち書きデータを利用し、平仮名 KyTea 単語分割システムを作成し、評価した。テストデータには 40 万文の Wikipedia のデータと 40928 文の BCCWJ による平仮名文の単語分かち書きデータをそれぞれ利用した。テストデータに利用した Wikipedia のデータは、BERT の

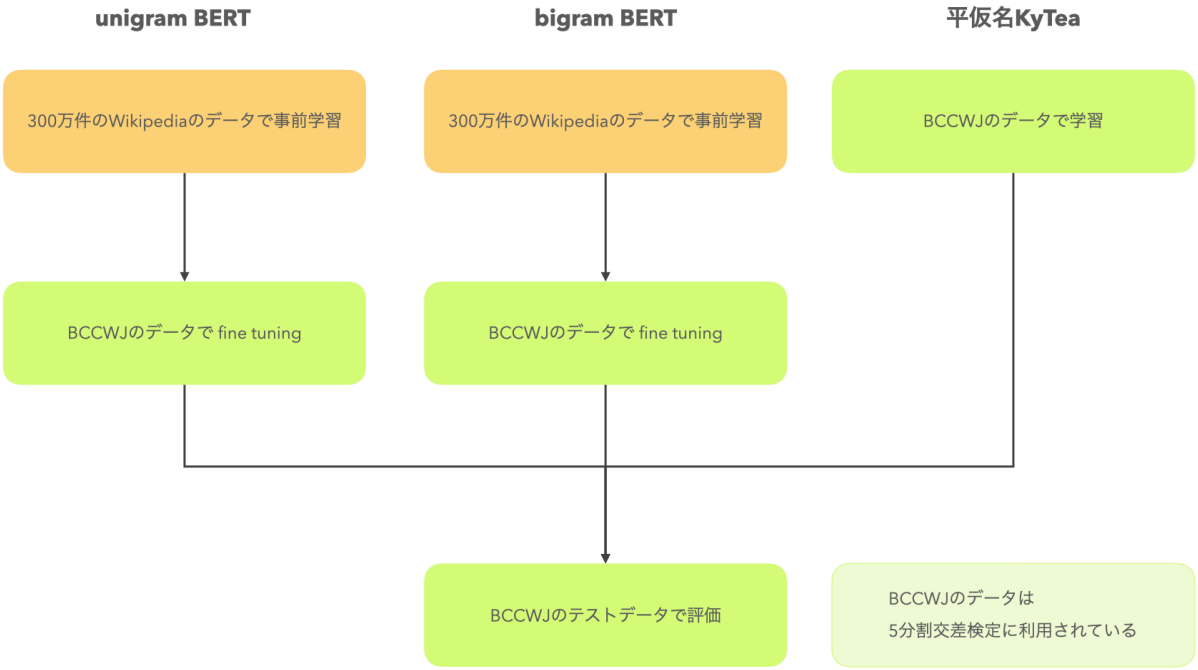

図 2 実験 1：BCCWJ によるファインチューニングの実験

事前学習およびファインチューニング用の学習データと重複がないデータである。この実験を可視化した図を**図 3** に示す。

実験 2 において BERT の事前学習で使用したパラメータおよびファインチューニングで使用したパラメータはどちらも実験 1 で利用したパラメータと同一である。

5.3 評価手法

作成する unigram BERT 単語分割システムと bigram BERT 単語分割システムは文字データを入力として与えると、入力に対してモデルが推定した、平仮名文を分割するかどうかの 0 と 1 のタグ情報を出力する。入力として与えるデータの形式は unigram BERT 単語分割システムに対しては unigram の形式であり、bigram BERT 単語分割システムに対しては bigram の形式である（表 2）。unigram BERT 単語分割システムと bigram BERT 単語分割システムに対して、テストデータを入力として与え、モデルから出力されるタグの一致率を正答率として評価する。

平仮名 KyTea 単語分割システムに関しては平仮名のみで構成された文字データを入力として与えるとモデルが推定する単語分割情報が出力される。出力された単語分割情報に対し単語分割の先頭を「1」に、それ以降を「0」と変換する。そして予め作成しておいたタグの正解データとの一致率を正答率として評価する。

6. 実験結果

まず、実験 1：BCCWJ によるファインチューニングの実験における各システムに対する 5 分割交差検定の結果を**表 7** に示す。

表 7 から unigram BERT 単語分割システムは平仮名 KyTea 単語分割システムと比較し精度が 1.84 point 向上していることがわかる。また bigram BERT 単語分割システムは平仮名 KyTea 単語分割システムと比較して精度が 0.61 point 向上している。さらに unigram BERT 単語分割システムと bigram BERT 単語分割システムの精度を比較すると、unigram BERT 単語分割システムの方が bigram BERT 単語分割システムよりも精度が 1.23 point 上昇している。

次に、実験 2：Wikipedia によるファインチューニングの実験の結果を**表 8** に示す。

表 8 実験 2：Wikipedia によるファインチューニングの実験における各モデルの精度

	Wikipedia	BCCWJ
unigram BERT 単語分割システム	**99.32**	**95.65**
bigram BERT 単語分割システム	99.08	95.36
平仮名 KyTea 単語分割システム	97.87	94.00

表 8 から unigram BERT 単語分割システムは、平仮名 KyTea 単語分割システムと比較し、Wikipedia をテスト

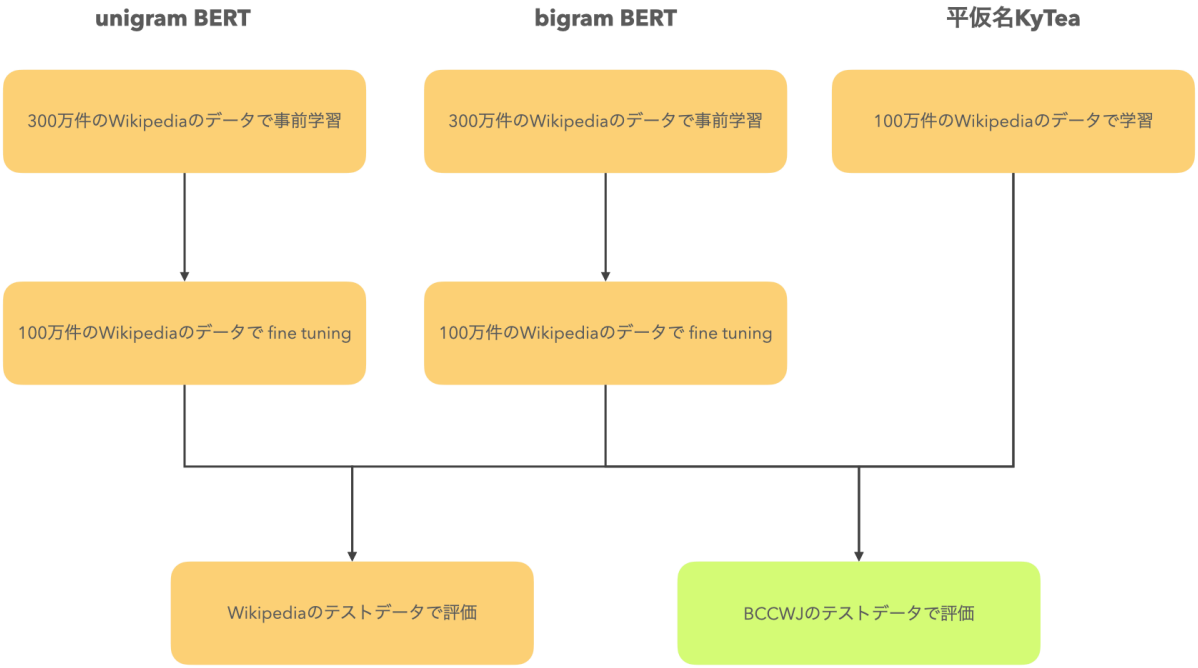

図 3 実験 2：Wikipedia によるファインチューニングの実験

表 7 実験 1：BCCWJ によるファインチューニングの実験における各モデルの精度

	unigram BERT 単語分割システム	bigram BERT 単語分割システム	平仮名 KyTea 単語分割システム
1 / 5	97.48	96.00	97.07
2 / 5	97.60	96.31	**97.21**
3 / 5	97.95	**97.02**	96.68
4 / 5	97.36	96.11	91.13
5 / 5	**97.98**	96.75	97.06
平均	**97.67**	96.44	95.83

データとした場合は精度が 1.45 point 向上し、BCCWJ のコアデータをテストデータとして利用した場合は精度が 1.65 point 向上していることが分かる。また bigram BERT 単語分割システムは、平仮名 KyTea 単語分割システムと比較し、Wikipedia をテストデータとした場合は精度が 1.21 point 向上し、BCCWJ をテストデータとして利用した場合には精度が 1.36 point 向上していることが分かる。さらに、unigram BERT 単語分割システムと bigram BERT 単語分割システムの精度を比較すると、unigram BERT 単語分割システムの方がより高い精度であった。精度の差は、Wikipedia をテストデータとした場合は 0.24 point であり、BCCWJ をテストデータとした場合は 0.29 point であった。

7. 考察

表 7 より、実験 1 において、作成した 2 種類の平仮名 BERT 単語分割システムの精度は、平仮名 KyTea 単語分割システムの精度よりも高いことが確認できる。さらに表 8 より、実験 2 においても、作成した 2 種類の平仮名 BERT 単語分割システムの精度が平仮名 KyTeay 単語分割システムの精度よりも高いことが確認できる。これにより、unigram BERT 単語分割システムと bigram BERT 単語分割システムは有効であるといえる。

また、表 7 と表 8 両方において、unigram BERT 単語分割システムと bigram BERT 単語分割システムの精度を比較すると、unigram BERT 単語分割システムの精度は bigram BERT 単語分割システムの精度よりも高いことが確認できる。bigram の方が unigram より情報量が多くなるため、我々は bigram BERT 単語分割システムの方が、unigram BERT 単語分割システムを上回ることを予想していたが、結果は逆であった。この理由としては、モデルの大きさに対応して必要になる学習データの差が考えられる。本研究において使用した平仮名 BERT の語彙数は unigram BERT 作成では 300 であり、bigram BERT 作成では 80956 であった。つまり bigram BERT 作成において使用した語彙数の方が unigram 作成において使用した語彙

数より約 270 倍多い。その分、モデルは大きくなるため、必要な学習データも多くなると考えられる。ところが、2 種類の平仮名 BERT 単語分割システムにおける事前学習で利用したデータ数はどちらも 300 万文であった。つまり、モデルの大きさに必要な学習データ量に対して bigram BERT には十分な学習データではなかった可能性があり、それが unigram BERT 単語分割システムの精度が bigram BERT の精度を上回った要因であると考えられる。

次に、実験 1 と実験 2 の結果を比較する。BCCWJ をテストデータにした実験結果同士（表 7 と表 8 の BCCWJ の結果）を比較すると、実験 1 における 2 種類の平仮名 BERT 単語分割システムの精度は、実験 2 における 2 種類の平仮名 BERT 単語分割システムの精度よりも高いことが分かる。これは、実験 1 のファインチューニングのデータはテストデータと同じ BCCWJ であるが、実験 2 では Wikipeda のデータを利用しているためであると考えられる。特に、BCCWJ では正確な読みと単語分割の区切りの情報を利用しているが、Wikipedia は疑似データであるため、Wikpiedia データの質は BCCWJ よりも低いと考えられる。ここで、実験 1 で利用した BCCWJ は約 4.5 万件であるのに対して実験 2 で利用した Wikipedia のデータは 100 万文であることを考えると、ファインチューニングにおける疑似データの量を増やしても、テストデータと同ドメインの正確なデータには及ばないことが分かる。

一方で、大量の Wikipedia の疑似データを与えた際、同 Wikipedia のテストデータに対する正解率は 99%を超える（表 8）。そのため、テストデータと同じドメインで、なおかつテストデータと整合性のある単語分割の情報をもつ大量のデータを利用してファインチューニングした場合には、かなり高い精度で単語分割が行えることが分かる。

最後に、本研究で利用した BERT 作成用の事前学習のデータ量は 300 万件であるが、この量を増やすことで、平仮名 BERT 単語分割システムの精度が向上する可能性がある。この点に関しては今後の課題と考えている。

8. おわりに

本研究では、平仮名文に特化して学習した 2 種類の BERT を利用した文単語分割システム、unigram BERT 単語分割システムと bigram BERT 単語分割システムを作成した。BERT の事前学習には、MeCab を利用して Wikipedia の平仮名文のデータから作成した文字 unigram または文字 bigram のデータを利用し、平仮名文の単語分かち書きのデータでファインチューニングを行うことで作成したものである。BCCWJ のコアデータを利用した五分割交差実験と、Wikipedia の疑似データを利用したファインチューニングによる学習に対し、Wikipedia および BCCWJ をテストデータにした実験において、これらの平仮名文の単語分割の精度は共に KyTea を用いた平仮名文単語分割システムの精度を上回った。また、unigram BERT 単語分割システムと bigram BERT 単語分割システムの精度を比較すると、unigram BERT 単語分割システムの方が精度が向上した。これにはモデルの大きさに対する事前学習のデータ数が影響したと考えられる。

また、実験により、ファインチューニングに利用するデータは、大量のドメインの異なった疑似データよりも、少量のドメインの等しい、テストデータと整合性の取れたデータの方がよいことが分かった。

謝辞 本研究は 2022 年度国立情報学研究所公募型共同研究（22FC04）と JSPS 科研費 18K11421 の助成を受けています。

参考文献

[1] 井筒順, 明石陸, 加藤涼, 岸野望叶, 小林汰一郎, 金野佑太, 古宮嘉那子 *MeCab* による平仮名のみの形態素解析, 言語処理学会　第 26 回年次大会　発表論文集（2020）
[2] Jun Izutsu, Kanako Komiya *Morphological Analyzer Using the Bi-LSTM Model Only for Japanese Hiragana Sentences*, International Journal on Natural Language Computing, vol. 11, no. 1, (2022.02)
[3] 工藤拓, 市川宙, David Talbot, 賀沢秀人 *Web* 上のひらがな交じり文に頑健な形態素解析, 言語処理学会　第 18 回年次大会　発表論文集（2012）
[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova *BERT: Pre-training of deep bidirectional transformers for language under- standing*. In Proceedings of the 2019 Conference of the North American Chapter of the Associa- tion for Computational Linguistics: Human Lan- guage Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186, 2019.
[5] 林 聖人, 山村 毅ひらがな語の追加と形態素解析の精度についての考察析, 愛知県立大学情報科学部平成 28 年度卒業論文要旨（2017）
[6] 鈴木雅弘, 坂地泰紀, 和泉潔, 石川康金融文書を用いた追加事前学習言語モデルの構築と検証, 言語処理学会　第 22 回年次大会　発表論文集（2022）
[7] 森山柊平, 大野誠寛, 増田英考 , 絹川博之 *Recurrent Neural Network Language Model* を用いたべた書きかな文の形態素解析, 情報処理学会論文誌 59 (10), 1911-1921, 2018-10-15
[8] 森山柊平, 大野誠寛 *RNN* とロジスティック回帰を用いた平仮名文の逐次的な形態素解析, 自然言語処理（2022）
[9] Makawa Kikuo, Yamazaki Makoto, Ogiso Toshinobu, Maruyama Takehiro, Ogura Hideki, Kashino Wakako, Koiso Hanae, Yamaguchi Masaya, Tanaka Makiro, Den Yasuharu(2014). *Balanced corpus of contemporary written Japanese*. Balanced corpus of contemporary written Japanese. LREC2014 345–371