# 平仮名 BERT を用いた平仮名文の分割

井筒順 [1]　古宮嘉那子 [2]　新納浩幸 [3]
[1] 茨城大学大学院理工学研究科情報工学専攻 [2] 東京農工大学工学研究院
[3] 茨城大学大学院理工学研究科情報科学領域
21nm707h@vc.ibaraki.ac.jp　kkomiya@go.tuat.ac.jp
hiroyuki.shinnou.0828@vc.ibaraki.ac.jp

## 概要

既存の日本語の形態素解析システムの性能は非常に高いが、これら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単語分割することは難しい。本論文では漢字以外の文字、つまり平仮名、数字、記号からなる文字ベースの unigram BERT と bigram BERT を作成し、これらを利用して二種類の平仮名文の単語分割システムを構築した。これらを KyTea を利用した平仮名分割システムと比較したところ、unigram BERT による平仮名分割システムの性能が最も高くなったことを示す。

## 1　はじめに

日本語には MeCab[1) ]や Chasen[2)]等の形態素解析システムが存在している。これらのシステムの性能は非常に高いが、漢字仮名混じりの文を対象にしているため、ほとんど全てが平仮名で書かれた文[3)]を単語分割することは難しい。

日本語初学者が最初に習得するのは平仮名であり平仮名で構成された文章を読む機会が多い。しかし平仮名で構成された文章を滞ることなく読むことは日本語の母語話者であっても難しい。

本稿では BERT を利用した 2 種類の平仮名文の分割システムを作成しその性能評価を行う。さらに、KyTea[4)]を用いた平仮名文 KeyTea 単語分割システムを作成し、作成した 2 種類の BERT による平仮名文の単語分割システムと比較する。

## 2　関連研究

工藤ら [1] 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性能を向上させる手法を提案している。また、林ら [2] 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 [3] 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 [4] 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回、学習とファインチューニングを行うことで形態素解析の性能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 [5] は RNNLM を用いたべた書きかな文の形態素解析を行ない、その性能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 [6] は 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手法を提案し、平仮名文における形態素解析の性能向上とシステムの高速化を報告している。

分野に特化した BERT を作成する研究として代表的なものには鈴木ら [7] の研究がある。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。鈴木らは、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。

本論文は井筒 [8] を再実験し、追加実験と考察を加えたものである。

## 3　提案手法

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

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）http://www.phontron.com/kytea/

デルのうち、1 つ目のモデルは unigram BERT モデルである。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。我々は上記 2 つのモデル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。本研究では、これら 2 つの平仮名文の単語分割システムの F 値を比較する。さらに、KyTea による平仮名文の単語分割のモデルを作成し、上記 2 つの平仮名文の単語分割システムの F 値と比較する。

### 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。この unigram BERT に対し、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。

### 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。この bigram BERT に対し、平仮名文の単語分割の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。

## 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 を作成するための事前学習用のデータとして、Wikipedia [5)6)] を利用した。

本データの作成方法は以下である。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。MeCab の読みデータから作成しているため、出力される平仮名文は、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。ただし、bigram の終端における文字列は、unigram との文字数を揃えるために、句点に文字種「*」を追加した「。*」としている。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。

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。

### 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、Wikipedia による平仮名文の単語分かち書きデータを作成した。本データは、MeCab の単語分割結果を信じ、その平仮名表記を利用して作成した。また、単語分割を二値分類として処理するため、分割対象位置の文字を 1、それ以外の文字を 0 とするタグデータを作成した。

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。

### 4.3 日本語書き言葉コーパスによる平仮名文の単語分かち書きデータ

日本語書き言葉コーパス（以下 BCCWJ[9] と記す）のコアデータは人手により単語に分けられたデータであるので、これ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用のデータおよびテストデータとして利用する。

本データの作成方法は以下である。まず BCCWJ のコアデータの読み情報を利用し、ほぼ平仮名で構成された文に変換した。次にそれらの文を文字 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用の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用のデータとして利用する。また BCCWJ のコアデータの単語区切りを利用し単語分割を二値分類として処理するための、分割対象位置の文字を 1、それ以外の文字を 0 とするタグ

5） https://dumps.wikimedia.org/jawiki/latest/
6） jawiki-latest-pages-articles.xml.bz2

データを作成した。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。

unigram BERT 作成及び bigram BERT 作成において使用した語彙の総数はそれぞれ 300 と 80956 である。語彙には平仮名、片仮名、アルファベット、数字、複数の記号が含まれている。

# 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 F 値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。

また二つの提案手法と比較するために、KyTea を用いて平仮名文の単語分割システムを作成した。これを以降、平仮名 KyTea 単語分割システムと呼ぶ。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。学習には、二つの提案手法による単語分割システムのファインチューニングに利用したデータと同内容のものを用いた。ただし、フォーマットは KyTea に合うように変形した。

作成した単語分割システムは、テストデータに対する正解率、適合率、再現率、F 値を評価した。

## 5.1 実験 1：BCCWJ によるファインチューニングの実験

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、2 種類の BERT 単語分割システムの F 値を平仮名 KyTea 単語分割システムの F 値と比較する。300 万文の Wikipedia による事前学習用データを利用して平仮名 BERT を作成し、40928 文の BCCWJ による平仮名文の単語分かち書きデータを用いて単語分割システムの 5 分割交差検定を行った。また平仮名 KyTea 単語分割システムについても、2 種類の BERT 単語分割システムと同様のデータを用いて、5 分割交差検定を行った。ただし、平仮名 KyTea 単語分割システムには、事前学習した BERT を用いていない。

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 1 に、ファインチューニングで使用したパラメータを表 2 に示す。

**表 1** 事前学習におけるパラメータ

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

**表 2** ファインチューニングにおけるパラメータ

| | | | |
|---|---|---|---|
| ラベル数 | 12 | エポック数 | 50 |
| 学習率 | 1e-5 | | |

## 5.2 実験 2：Wikipedia によるファインチューニングの実験

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、3 つの単語分割システムの F 値を比較する。300 万文の Wikipedia による事前学習用データを利用して平仮名 BERT を作成し、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て単語分割システムのファインチューニングを行った。なお、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、unigram BERT 単語分割システムおよび bigram BERT 単語分割システムに利用した 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。

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 epoch 数以外は実験 1 で利用したパラメータと同一である。実験 2 における epoch 数は 24 とした。

# 6 実験結果

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 分割交差検定の正解率、適合率、再現率、F 値を表 3 に示す。

**表 3** 実験 1 における各システムの結果

| | 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 |
|---|---|---|---|
| 正解率 | **97.74** | 96.98 | 95.83 |
| 適合率 | **94.36** | 92.56 | 90.93 |
| 再現率 | **94.24** | 92.60 | 88.56 |
| F 値 | **94.30** | 92.58 | 89.66 |

表 3 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、F 値が 4.64 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て F 値が 2.92 point 向上している。さらに

unigram BERT 単語分割システムは bigram BERT 単語分割システムよりも F 値が 1.72 point 高い。

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 4 に示す。

**表 4** 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システムの結果

| テストデータを Wikipedia にした場合 | | | |
|---|---|---|---|
| | 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 |
| 正解率 | **99.32** | 99.08 | 97.17 |
| 適合率 | **98.14** | 97.41 | 92.83 |
| 再現率 | **97.83** | 97.15 | 91.76 |
| F 値 | **97.98** | 97.28 | 92.29 |
| テストデータを BCCWJ にした場合 | | | |
| | 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 |
| 正解率 | **95.65** | 95.36 | 93.96 |
| 適合率 | **90.85** | 89.94 | 86.68 |
| 再現率 | **86.67** | 85.93 | 81.72 |
| F 値 | **88.71** | 87.89 | 84.12 |

表 4 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 をテストデータとした場合は F 値が 5.69point 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 F 値が 4.59point 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 F 値が 4.99point 向上し、BCCWJ をテストデータとして利用した場合には F 値が 3.77point 向上していることが分かる。さらに、unigram BERT 単語分割システムは bigram BERT 単語分割システムより高い F 値となった。F 値の差は、Wikipedia をテストデータとした場合は 0.70point であり、BCCWJ をテストデータとした場合は 0.82point であった。

## 7 考察

表 3 と表 4 より、実験 1 と実験 2 でそれぞれ作成した 2 つの平仮名 BERT 単語分割システムの F 値は、平仮名 KyTea 単語分割システムの F 値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 単語分割システムが有効であるといえる。

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 3 と表 4 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 F 値の方が高い。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 は疑似データゆえに、Wikipedia データの質は BCCWJ よりも低いと考えられる。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 4）。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、かなり単語分割の評価値が高くなることが分かる。

## 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。このシステムは BERT の事前学習として、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 のデータを利用したファインチューニングによる実験において、平仮名文の単語分割の F 値は共に KyTea を用いた平仮名文単語分割システムの F 値を上回った。また、unigram BERT 単語分割システムと bigram BERT 単語分割システムの F 値を比較すると、unigram BERT 単語分割システムの方が F 値が向上した。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] 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, pp. 1272–1275, 2012.

[2] 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.

[3] 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, pp. 65–69, 2020.

[4] Jun Izutsu and Kanako Komiya. Morphological analysis of japanese hiragana sentences using the bi-lstm crf model,. **10th International Conference on Natural Language Processing (NLP 2021)**, 2021.

[5] 森山柊平, 大野誠寛, 増田英孝, 絹川博之ほか. 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, Vol. 59, No. 10, pp. 1911–1921, 2018.

[6] 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, Vol. 29, No. 2, pp. 367–394, 2022.

[7] 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 28 回年次大会発表論文集, pp. 588–592, 2020.

[8] 井筒順, 古宮嘉那子, 新納浩幸. 平仮名 bert による平仮名文の分割. 研究報告自然言語処理（NL）, Vol. 2022-NL-253, No. 1, pp. 1–7, 2022.

[9] Kikuo Maekawa, Makoto Yamazaki, Toshinobu Ogiso, Takehiko Maruyama, Hideki Ogura, Wakako Kashino, Hanae Koiso, Masaya Yamaguchi, Makiro Tanaka, and Yasuharu Den. Balanced corpus of contemporary written japanese. **Language resources and evaluation**, Vol. 48, No. 2, pp. 345–371, 2014.

## A　追加実験

実験 1 と実験 2 の結果において、2 種類の平仮名 BERT 単語分割システムの F 値を比較すると、unigram BERT 単語分割システムの F 値がより良い結果であることが確認できる。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。つまり語彙数に約 270 倍の差が存在する。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 F 値が bigram BERT の F 値を上回った要因であると考えられる。

語彙数に大幅な差があることに着目し、我々は、実験 1 のテストデータから記号等の文字種を除いたデータを利用し、各システムの評価を再度算出した。実験 1 のテストデータから取り除かなかった文字種は、平仮名・片仮名・句点・読点・長音・符空白である。実験 1 のテストデータの各文に対して上記以外の文字種が含まれる文は評価せず、上記のみの文字種で構成された文を各システムに入力し、評価した。本追加実験における結果を表 5 に示す。

**表 5**　実験 1 においてテストデータから記号を除いた場合の各システムの結果

| | 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 |
|---|---|---|---|
| 正解率 | **98.06** | 97.41 | 96.45 |
| 適合率 | **95.04** | 93.38 | 92.15 |
| 再現率 | **94.93** | 93.50 | 89.81 |
| F 値 | **94.99** | 93.44 | 90.91 |

表 5 より文字種を制限することにより全体的に F 値が向上していることが確認できる。特に、unigram BERT 単語分割システムと bigram BERT 単語分割システムにおける F 値の差は 1.55point である。表 3 における 2 つの単語分割システムの F 値の差が 1.72 であったことから、テストデータにおける文字種を制限することで F 値の差が縮まっていることがわかる。

## B　実験の可視化

実験 1 におけるシステム構築を可視化した図を図 1 に示す。また、実験 2 におけるシステム構築を可視化した図を図 2 に示す。

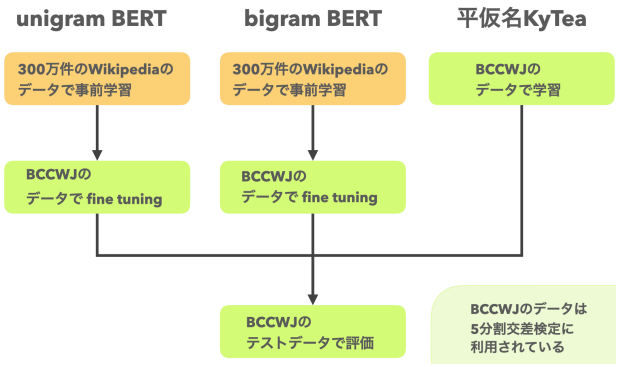


**図 1**　実験 1：BCCWJ によるファインチューニングの実験

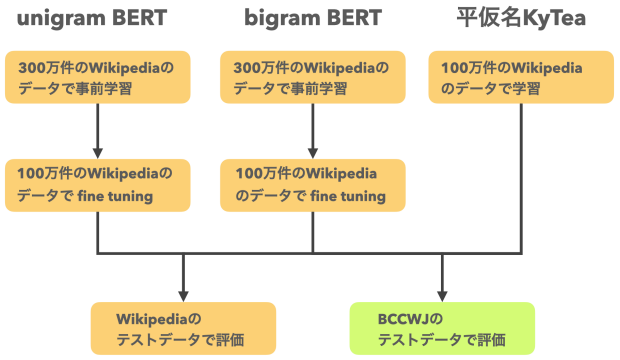


**図 2**　実験 2：Wikipedia によるファインチューニングの実験