

学修番号 14890524

## 修士論文

日本語学習者文の頑健な単語分割のための分野適応

塘 優旗

2016 年 2 月 23 日

首都大学東京大学院  
システムデザイン研究科 情報通信システム学域

塘 優旗

審査委員：

小町 守 准教授 (主指導教員)

山口 亨 教授 (副指導教員)

高間 康史 教授 (副指導教員)

# 日本語学習者文の頑健な単語分割のための分野適応\*

塘 優旗

## 内容梗概

近年、日本語の学習者の増加とともに自然言語処理を利用した作文誤り検出・訂正を行うことが求められてきている。既存の誤り訂正手法の多くは事前に単語分割を行う必要があり、水本らの統計的機械翻訳の手法を用いた日本語学習者の作文自動誤り訂正においては、正しく単語分割できた場合は訂正の精度が高くなることが述べられている。しかしながら、日本語学習者の文は、うまく文字の変換がされていない場合や、誤りを含むなどの理由から、既存の単語分割器や形態素解析器では単語分割に失敗しやすい。

本研究では上記のような誤りを含む日本語学習者の日本語文に対して頑健な単語分割を行うことを目標とする。現在、日本語単語分割の手法として主に利用されているのは、ルールベースのものや、機械学習に基づくものである。これらを上記のようなテキストに分野適応するには誤りや表記揺れに対応したたくさんのルールを人の手で定めることや、分野適応先の文に対して単語分割のアノテーションが行われた大量のコーパスを作る必用があり高コストである。そこで、本研究では予め大量の一般的なコーパスで学習を行い、一部のみアノテーションされた分野適応先のコーパスで追加学習を行う機械学習を用いた手法を提案する。

1つ目の手法としてアノテーションが曖昧な部分に関しては周辺尤度を用いて学習を行う条件付き確率場 (CRF) の拡張を利用し言語学習 SNS Lang-8 中における日本語学習者の文に対して分野適応を行う。予め様々な分野のテキストにアノテーションのされたコーパスである現代日本書き言葉均衡コーパス (BCCWJ) を用いて学習し、Lang-8 から抽出した日本語学習者の文と添削文のペアから一部のみアノテーションされた訓練データを自動で作成し追加学習することで分野適応する。

2つ目の手法として深層ニューラルネットを用いて日本語学習者文の分野適応を

---

\*首都大学東京大学院 システムデザイン研究科 情報通信システム学域 修士論文, 学修番号 14890524, 2016 年 2 月 23 日.

行う。予め日本語学習者の文を利用し、word2vec の手法を用いてシステムに入力される文字の分散表現を訓練し、それらの分散表現を初期値として上記の BCCWJ を用いて深層ニューラルネット全体の訓練を行うことで分野適応を行う。また、CRF の拡張の場合と同様に部分的なアノテーションのされた訓練データを用いた分野適応も行った。

これら 2 つの手法に対して実験を行い、分野適応の際に利用する訓練データの利用の仕方、適切なパラメータ設定により精度が向上することを示す。特に CRF の拡張を用いた手法においては、分野適応の訓練時に利用する文を学習者文と添削文間での挿入、削除数に関して制限することによって学習者テキストの単語分割精度の向上が見られた。また、手法間での出力結果を交えて考察を行う。

本論文の構成は以下のようにになっている。第 1 章では本研究全体の提案手法の概要、貢献を述べる。第 2 章では部分的アノテーションを利用した日本語学習者文の単語分割について関連研究、提案手法、実験、考察について述べる。第 3 章では深層ニューラルネットを利用した日本語学習者文の単語分割について関連研究、提案手法、実験、考察について述べる。最後に第 4 章では本研究のまとめ、今後の展望について述べる。

# Domain Adaptation For Robust Word Segmentation of Japanese Learner's Text\*

Yuki Tomo

## Abstract

In recent years, error correction systems for Japanese learner's text using natural language processing have become necessary with increasing number of Japanese language learners. Most of error correction systems require word segmented sentences. One of the previous studies used statistical machine translation for error correction and showed that correctly word segmented sentences are corrected easily. However, most of learner's sentences have character conversion and spelling errors. Therefore, word segmentation for learner's text is very difficult for existing word segmentation and morphological analysis systems.

The purpose of this study is to introduce robust word segmentation for Japanese Learner's text including such errors. In these days, most of Japanese word segmentation systems are based on rules or machine learning technics. These systems need many rules or large annotated corpus for target domain to adapt to such sentences. Additionally, making these rules and corpus are very costly. For these reasons, we use standard Japanese word segmented corpus and partially annotated corpus for target domain to train machine learning systems.

First, we use the extension of Conditional Random Fields (CRF) that models marginal probabilities over partially annotated data to adapt word segmentation model for Japanese learner's texts obtained from a free language-exchange social network Lang-8. Initially we train the model using popular Japanese fully annotated corpus called BCCWJ. Next, we retrain the model using partially annotated corpus constructed from the pairs of learner's and corrected sentences

---

\*Master's Thesis, Department of Information and Communication Systems, Graduate School of System Design, Tokyo Metropolitan University, Student ID 14890524, February 23, 2016.



for domain adaptation.

Second, we use Deep Neural Network for domain adapted word segmentation. Using the word2vec method, we make distributed representation of characters (character embedding) from Japanese learner's texts. We use the character embedding as initial value and train Deep Neural Network using BCCWJ corpus. In addition, we retrain the model using partially annotated corpus.

We show that using restricted training data and appropriate parameters improves word segmentation of Japanese learner's text. Especially in the extension of the CRF, restricting the sentences in training for domain adaptation with the numbers of insertion and deletion between corrected sentence and learner's sentence improve the accuracy of word segmentation in learner's text. Furthermore, we compare the results of these methods and discuss the differences.

This paper is organized as follows. Chapter 1 shows the outline of the proposed methods and the contribution of this study. Chapter 2 describes related work, proposed method, experiments, and discussion of the first method that the extension of CRF using partially annotated corpus. Chapter 3 explains related work, proposed method, experiments, and discussion of the second method using Deep Neural Network. Chapter 4 presents conclusion and future work.

# 目次

図目次	vii
表目次	viii
第 1 章 はじめに	2
第 2 章 部分的アノテーションを利用した CRF による日本語学習者文の単語分割	4
2.1 導入	4
2.2 関連研究	5
2.2.1 単語分割, 形態素解析のドメイン適応	5
2.2.2 日本語学習者の作文の誤り訂正に向けた単語分割	6
2.2.3 部分的アノテーションを利用した条件付き確率場	8
2.3 部分的アノテーションを用いた日本語学習者文の単語分割	10
2.3.1 問題設定	10
2.3.2 単語分割基準	10
2.4 実験	11
2.4.1 ベースライン	11
2.4.2 データセット	12
訓練用データ	12
テストデータ	13
2.4.3 素性テンプレート	14
2.4.4 評価手法	14

2.4.5	ツール . . . . .	15
2.4.6	実験結果 . . . . .	15
2.5	考察 . . . . .	17
2.5.1	挿入数, 削除数の影響 . . . . .	17
2.5.2	KyTea との比較 . . . . .	20
2.5.3	MeCab との比較 . . . . .	22
2.6	まとめと今後の課題 . . . . .	23
<b>第 3 章</b>	<b>深層ニューラルネットを利用した乱れた日本語の頑健な単語分割</b>	<b>24</b>
3.1	導入 . . . . .	24
3.2	関連研究 . . . . .	24
3.3	提案手法 . . . . .	25
3.3.1	分散表現 . . . . .	25
3.3.2	ニューラルモデルを利用した日本語単語分割 . . . . .	26
3.3.3	訓練 . . . . .	29
3.4	実験 . . . . .	30
3.4.1	ツール, パラメータセット . . . . .	30
3.4.2	実験結果 . . . . .	31
3.5	考察 . . . . .	31
	B-LSTM, P-CRF 間の比較 . . . . .	32
	B-LSTM (BCCWJ), B-LSTM (BCCWJ + Lang-8 (char2vec)) 間の比較 . . . . .	32
	B-LSTM (BCCWJ), B-LSTM (BCCWJ + ins1del0 (re- train)) 間の比較 . . . . .	33
3.6	まとめと今後の課題 . . . . .	33
<b>第 4 章</b>	<b>おわりに</b>	<b>41</b>
4.1	今後の展望 . . . . .	42
	<b>参考文献</b>	<b>44</b>
	<b>発表リスト</b>	<b>46</b>



# 図目次

3.1	ニューラル単語分割モデル . . . . .	27
3.2	LSTM メモリーユニット . . . . .	28

# 表目次

2.1	各データセットの文数 . . . . .	13
2.2	日本語学習者文の単語分割における各手法および訓練データの比較	16
2.3	P-CRF (BCCWJ), P-CRF (BCCWJ+ins1del0) の削除誤り箇所 の単語分割改善例 . . . . .	17
2.4	P-CRF (BCCWJ), P-CRF (BCCWJ+ins1del0) の半角文字箇所 の単語分割改善例 . . . . .	18
2.5	P-CRF (BCCWJ), P-CRF (BCCWJ+ins1del0) 間の単語分割 悪化例 . . . . .	18
2.6	ins1del0 の学習者文と添削文のペア . . . . .	18
2.7	P-CRF (BCCWJ+ins1del0), P-CRF (BCCWJ+ins2del0) 間の 単語分割悪化例 . . . . .	19
2.8	P-CRF (BCCWJ), P-CRF (BCCWJ+ins0del1) 間の単語分割悪 化例 . . . . .	19
2.9	ins0del1 中の学習者文と添削文のペア . . . . .	20
2.10	KyTea (BCCWJ+ins1del0), P-CRF (BCCWJ+ins1del0) 間の 単語分割改善例 . . . . .	20
2.11	KyTea (高性能 SVM モデル), P-CRF (BCCWJ+ins1del0) 間の 単語分割改善例 . . . . .	21
2.12	MeCab, P-CRF (BCCWJ+ins1del0) 間の単語分割改善例 . . .	22
2.13	MeCab, P-CRF (BCCWJ+ins1del0) 間の単語分割悪化例 . . . .	23
3.1	パラメータ . . . . .	31

3.2	日本語学習者文の単語分割における各手法および訓練データの比較	35
3.3	P-CRF (BCCWJ), B-LSTM (BCCWJ) の単語分割悪化例 - 誤り箇所, ひらがな箇所	36
3.4	P-CRF (BCCWJ), B-LSTM (BCCWJ) の単語分割改善例 - 誤り箇所, ひらがな箇所	36
3.5	P-CRF (BCCWJ), B-LSTM (BCCWJ) の単語分割改善例 - アルファベット, カタカナ語の複合語	36
3.6	B-LSTM (BCCWJ), B-LSTM (BCCWJ+Lang-8 (char2vec)) の単語分割悪化例 - アルファベット, カタカナ	37
3.7	B-LSTM (BCCWJ), B-LSTM (BCCWJ+Lang-8 (char2vec)) の単語分割悪化例 - 誤りが含まれていない箇所	37
3.8	B-LSTM (BCCWJ), B-LSTM (BCCWJ+Lang-8 (char2vec)) の単語分割悪化例 - 誤りを含む箇所	38
3.9	B-LSTM (BCCWJ), B-LSTM (BCCWJ+Lang-8 (char2vec)) の単語分割改善例 - 誤りを含む箇所	38
3.10	B-LSTM (BCCWJ), B-LSTM (BCCWJ+Lang-8 (char2vec)) の単語分割改善例 - 漢字変換無し (ひらがな) 箇所	39
3.11	B-LSTM (BCCWJ), B-LSTM (BCCWJ+ins1del0 (retrain 学習率 $\alpha = 0.00001$ )) の単語分割改善例	39
3.12	B-LSTM (BCCWJ), B-LSTM (BCCWJ+ins1del0 (retrain 学習率 $\alpha = 0.00001$ ))) の単語分割悪化例	40

## 第1章 はじめに

近年、日本語の学習者の増加とともに自然言語処理を利用した作文誤り検出・訂正を行うことが求められてきている。既存の誤り訂正手法の多くは事前に単語分割を行う必要があり、水本ら [21] の統計的機械翻訳の手法を用いた日本語学習者の作文自動誤り訂正においては、正しく単語分割できた場合は訂正の精度が高くなることが述べられている。しかしながら、日本語学習者の文は、うまく文字の変換がされていない場合や、誤りを含むなどの理由から、既存の単語分割器や形態素解析器では単語分割に失敗しやすい。

本研究では上記のような表記の揺れを含む日本語学習者の日本語文といった新聞記事のように整っていない日本語文に対して頑健な単語分割を行うことを目標とする。現在、日本語単語分割の手法として主に利用されているのは、ルールベースのものや、機械学習に基づくものである。これらを上記のようなテキストに分野適応するには誤りや表記揺れに対応したたくさんのルールを人の手で定めることや、分野適応先の文に対して単語分割のアノテーションが行われた大量のコーパスを作る必用があり高コストである。そこで、本研究では大量の一般的なコーパスをメインのコーパスとし、分野適応先の一部がアノテーションされたコーパスもしくはアノテーションのされていない生コーパスを補助的に利用することで分野適応を行う機械学習を用いた手法を提案する。

本論文は主に2つの手法で課題に取り組む。

2章では、アノテーションが曖昧な部分に関しては周辺尤度を用いて学習を行う条件付き確率場 (CRF) の拡張を利用し言語学習 SNS Lang-8 中における日本語学習者の文に対して分野適応を行う。予め様々な分野のテキストにアノテーションのされたコーパスである現代日本書き言葉均衡コーパス (BCCWJ) を用いて学習し、Lang-8 から抽出した日本語学習者の文と添削文のペアから一部のみアノテーションされた訓練データを自動で作成し追加学習することで分野適応する。

3章では、深層ニューラルネットを用いて日本語学習者文の分野適応を行う。日本語学習者文への分野適応のために、アノテーションのされていない Lang-8 の日本語学習者文コーパスを用いて予めシステムに入力される文字、文字種の分散表現を学習し、その予め学習された分散表現を初期値として BCCWJ コーパスを用い

て訓練を行う。

これら2つの手法に対して実験を行い、分野適応の際に利用する訓練データの利用の仕方、適切なパラメータ設定により精度が向上することを示す。特に CRF の拡張を用いた手法では分野適応の訓練時に利用する文を学習者文と添削文間での挿入、削除数によって制限することによって学習者テキストの単語分割精度の向上が見られた。また、手法間での出力結果を交えて考察を行う。

本研究の貢献として、以下の点があげられる。

- 日本語学習者文に対応した頑健な単語分割のためのコストの低い分野適応を提案
- 日本語学習者文の単語分割に関してのデータセットの作成、一致率の確認
- 条件付き確率場 (CRF) を利用し、一部のみアノテーションのされた日本語学習者文コーパスを訓練データとして利用する際に訓練用データの利用の仕方大きく結果が異なり、全体の精度が向上することを示す
- 深層ニューラルネットを用いて日本語学習者文の単語分割への分野適応を検討



## 第2章 部分的アノテーションを利用した CRF による 日本語学習者文の単語分割

### 2.1 導入

国際交流基金の「2012 年度 日本語教育機関調査」によると、海外の 136 の国と地域で、399 万人の人々が日本語を学習しており、日本語学習者の数は年々増加している。一方、日本語教師の数は 6.3 万人に止まり、全ての日本語学習者が十分な学習環境を得られているわけではない。日本語教師の数は 2009 年の調査結果から 28 % 増加しているが、日本語教師の不足を補うために、作文誤り検出・訂正などの自動添削を用いて日本語教師・学習者の支援をする必要性がある。

自然言語処理を利用して作文誤り検出・訂正を行うには、まず、単語分割を行う必要がある。たとえば、水本ら [21] の統計的機械翻訳の手法を用いた日本語学習者の作文自動誤り訂正においては、正しく分割できた場合は訂正の精度が高くなることが述べられている。しかしながら、日本語学習者の文は、うまく文字の変換がされていなかったり、誤りを含むなどの理由から、既存の単語分割器や形態素解析器では単語分割に失敗しやすい。また、今村ら [16] は、助詞誤りに限定し誤り訂正を行っているが、訓練の際に助詞誤りのみ含まれる学習者文と修正文のペアを用いているため、形態素解析で助詞をうまく分割できなかった場合は助詞の誤り訂正を行うことができない。したがって、日本語学習者の文をうまく単語分割できるようになることで、誤り訂正の精度向上に貢献できることが考えられる。

そこで、本章では日本語学習者の日本語文を、誤り訂正に適した形に単語分割することを目標とする。藤野ら [23] の先行研究を参考に、日本語学習者の書いた文と添削文のペアから添削が行われた箇所のみ単語分割のアノテーションを行い、その他の箇所に関しては曖昧なままにした学習者コーパスを作成する。藤野らは学習者コーパスを用いて KyTea を追加学習することで誤り箇所の分割精度は向上することができたが、誤っていない箇所も含めた全体の精度は低下することを報告している。一方、本研究ではアノテーションに曖昧な箇所が含まれていても系列全体を用いて訓練が行える条件付き確率場の拡張 [17] を利用することで訓練を行い、訓練用データの利用の仕方大きく結果が異なり、全体の精度が向上することを示す。また、それに際してテストデータとして日本語学習者文の単語分割に関するデータ

セットの作成を行い一致率を確認した。

## 2.2 関連研究

現在、日本語単語分割の手法として主に利用されているのは、ルールベースのもの\*や、機械学習に基づくもの [20] である。これらの単語分割の手法は、誤りを含んだ文や整っていない文に対して、単語分割の精度が落ちてしまう傾向がある。これは、誤りに対応したたくさんのルールを人の手で定めることや、誤りを含む文に対して単語分割のアノテーションが行われた大量のコーパスを作ることが困難なためである。

### 2.2.1 単語分割、形態素解析のドメイン適応

誤りを含む文と同様に、ドメイン適応のためのコーパスのアノテーションはとても高コストである。そこで、ドメイン特有の箇所のみアノテーションを行う**部分的アノテーション**がなされたコーパスの利用が考えられる。Neubig ら [11] は、日本語文の単語分割、読み推定タスクのドメイン適応のために部分的アノテーションにより作成されたコーパスを利用した点推定の手法を提案し、それらの実装である KyTea<sup>†</sup>を公開している。入力された文における、各単語境界、各単語に対しての読みは全て独立して推定されるため、一部の箇所のみアノテーションされたコーパスを訓練データとして利用することができる。彼らは、完全にアノテーションされた一般的なコーパスに加えて部分的アノテーションのされたコーパスを利用することで性能向上を報告している。部分的アノテーションを利用したコーパスの利用については本研究と同様であるが、コーパスのアノテーションを人手で行っている点と学習に用いる手法が異なる。

また、日本語学習者の文同様に、表記ゆれの大きい WEB 上の個人が書いたマイクロブログ等のテキストに対して頑健な形態素解析を行う研究も多くなされている。笹野ら [22] は、辞書中に存在するような正規語から派生する未知表記、未知オ

\*日本語形態素解析システム JUMAN <http://nlp.ist.i.kyoto-u.ac.jp/index.php?JUMAN>

<sup>†</sup><http://www.phontron.com/kytea/index-ja.html>



ノマトペを扱うために、単語の派生ルールとコストを人手で定め、形態素解析の際のラティス展開を拡張し性能向上を図っている。工藤ら [18] は、Web 上に頻出するひらがな交じり文に対して頑健な形態素解析を提案している。正規の文から各単語がひらがな化し、ひらがな混じり文が生成されるような過程をモデル化した生成モデルを導入し、大規模な WEB コーパス及び EM アルゴリズム [6] によってモデルのパラメータ推定を行い、ひらがな混じり文に対して頑健な形態素解析と正規文の導出を可能にしている。斉藤ら [19] は、工藤らの手法をひらがな化以外にも拡張し、単語の派生ルールを用いて正規語から辞書中に存在しない未知語に派生してしまう確率である崩れ表記語生成確率を大量の平文を用いてあらかじめ学習する。また、その崩れ表記語生成確率を素性として導入し、正解付きコーパスで学習を行っており性能向上が報告されている。これらの研究は、あらかじめ正規語からの派生パターンを与えているのに対して、本研究では直接コーパスから誤りの含まれる箇所に関しての単語分割を学習する点が異なる。

## 2.2.2 日本語学習者の作文の誤り訂正に向けた単語分割

藤野ら [23] は言語学習者向けの相互添削型 SNS である Lang-8<sup>‡</sup>から日本語学習者の文とそれに対応した添削文を取得し、文対間で単語分割のアノテーションを自動で行い学習者コーパスを作成することを提案している。彼らは、添削前後で変化のあった箇所の単語分割を利用するパターンを点推定の単語分割器である KyTea-0.3.2 [11] に再学習させることが学習者の文中の誤りを含む箇所における単語分割精度を高める一方、正解箇所の精度を低くすることを報告している。本研究では、藤野ら同様に学習者コーパスを作成し、添削前後で変化のあった箇所の単語分割のみアノテーションされた文を利用して、添削前後で削除、挿入が行われた学習者コーパス中の文の影響を確認し、条件付き確率場によって訓練することで、全体の評価値が向上するような学習者コーパスの利用制限を行うことができた。

学習者の文は、誤りを含んでいるので単語分割が失敗する傾向があるが、添削文に関しては学習者の文に対して誤り訂正などの添削が行われた文であるので、一般的な単語分割の手法でうまく単語分割が可能だと仮定する。このことから、添削文

---

<sup>‡</sup><http://lang-8.com/>

の単語分割を行い、その単語境界を学習者の文に反映することで大量の学習者の単語分割コーパスを作成する。

しかし、Lang-8 中の添削文の多くは、添削対象の文の言語が母語であるが必ずしも専門的知識を持たない一般のユーザの投稿である。そのため、学習者の文に複数の誤りが含まれていた場合に添削文中で訂正の不足が起こっていたり、曖昧な場合にはそのままになっていたりすることが起きうる。従って、比較的信頼することのできる誤り訂正が行われた箇所のみアノテーションを採用し、これ以外の箇所に關してはアノテーションを行わずに曖昧な状態のままとする。

このように、アノテーションを行うのが難しい箇所や曖昧な箇所にはアノテーションを行わずに、一部の箇所のみにアノテーションを行うことで 2.2.1 項で挙げた部分的アノテーションを実現する。以下のようなステップで行う。

#### 1. 添削前後の挿入、削除数の導出

学習者文と添削文のペア間において、文字の挿入・削除操作の箇所を動的計画法を用いて求める。各文字に対して、挿入 (Insertion) を I タグ、削除 (Deletion) を D タグ、操作なしを N で表すと以下ようになる。

**学習者の文**    でもじよ            ずじゃ    りません

**添削文**        でもじ    ようずじゃありません

**文字操作タグ** N N N I D D N N N D N N N N

#### 2. 添削文の単語分割

添削文に対して単語分割器で分割を行う。単語の開始文字を B タグ、単語の内部文字を I タグ、1 文字単語を S タグで表すと以下ようになる。

**学習者の文**    でもじよ            ずじゃ    りません

**添削文**        でもじ    ようずじゃありません

**単語分割 (添)** B I B    I I I B I B I B I S

**文字操作タグ**   N N N I D D N N N D N N N N

#### 3. 学習者文への単語分割の反映

添削文の分割箇所を学習者の文に反映する。挿入された箇所は、添削文において挿入文字の前と同じ単語を構成し、添削文で単語分割の終了文字となっていた場合、学習者文は挿入箇所の前で単語分割を行う。また、挿入文字の後ろと同じ単語を構成していた場合、学習者文はその単語の分割に従う。削

除箇所は，削除箇所の次の文字から別単語になっていた場合は，削除文字を削除箇所の前に接続する．また，削除箇所の前後の文字が同じ単語を構成していた場合はその単語の分割箇所での分割を行う．したがって以下のようなアノテーションとなる．

学習者の文    でもじよずじゃりません

単語分割 (学) B I B I I B I S B I S

また部分的アノテーションとした場合，添削が行われた箇所のみのアノテーションを行い曖昧なものを？タグとすると以下のようなタグ付けとなる．

学習者の文    でもじよずじゃりません

単語分割 (学) ? ? B I I ? ? S ? ? ?

### 2.2.3 部分的アノテーションを利用した条件付き確率場

部分的アノテーションを行い，曖昧なアノテーションを含む学習コーパスを利用した機械学習の手法として，坪井ら [17] は，アノテーションが曖昧な部分に関しては周辺尤度を用いて学習を行う条件付き確率場 (CRF) [9] の拡張を提案している．

はじめに，訓練の手法について説明を行う．入力列  $x = (x_1, x_2, \dots, x_T)$  を入力変数  $x_t \in X$  が要素となる列構造，ラベル列  $y = (y_1, y_2, \dots, y_T)$  をラベル変数  $y_t \in Y$  の列，  $\Phi(x, y) : X \times Y \rightarrow \mathbf{R}^d$  を入力列  $x$  とラベル列  $y$  の組から  $d$  次元の任意の素性ベクトルへの写像，  $\theta \in \mathbf{R}^d$  をモデルのパラメータベクトルとすると，一般的な CRF は  $x$  が与えられた時の  $y$  の条件付確率を式 2.21 でモデル化する．分母は正規化項である．

$$P_{\theta}(y|x) = \frac{e^{\theta \cdot \Phi(x,y)}}{Z_{\theta,x,y}} \quad (2.21)$$

例えば，文中の単語の品詞推定タスクを系列ラベリング問題として解く場合，入力列  $x$  は 1 文の単語列，ラベル列  $y$  が各単語に対しての品詞列となる．

次に  $y$  の一部が曖昧なデータの表現のために，  $L = (L_1, L_2, \dots, L_T)$  を入力  $x$  の各文字文字が取り得るラベル変数の値集合  $L_t \in 2^Y - \{\emptyset\}$  の列とする．例えば，



PennTreebank コーパス<sup>§</sup>における品詞に曖昧なアノテーションがされた文は以下のようなものがあり, “pending” の品詞は “VBG” または “JJ” としてアノテーションされている. このとき各ラベルはそれぞれ, DT: 限定詞, NN: 名詞単数, VBZ: 動詞 3 人称単数現在形, VBG: 動名詞または動詞現在分詞, JJ: 形容詞, SYM: 記号 である.

**入力文** That suit is pending .

$$L = (\{DT\}, \{NN\}, \{VBZ\}, \{VBG, JJ\}, \{SYM\}) \quad (2.22)$$

になる. 一般的な CRF では, このように  $y$  の一部だけが曖昧な  $L$  から直接学習することができないため,  $L$  に適合するあらゆるラベル列の集合を  $Y_L$  としたとき, 以下のようなモデルを用いる.

$$P_{\theta}(Y_L|x) = \sum_{y \in Y_L} P_{\theta}(y|x) \quad (2.23)$$

上記のモデルを用いることで, 部分的にアノテーションのされた訓練データを利用して CRF のパラメータの推定が可能になる.

次に, 訓練後のパラメータを利用して与えられた日本語文に対して単語分割のラベル付けを行う. 与えられた日本語文  $x$  に対して, 最も高い確率の高いラベル系列  $\hat{y}$  は以下のように定義され, ビタビアルゴリズムによって最適なラベル系列が探索される.

$$\hat{y} = \arg \max_{y \in \mathcal{Y}} P(y|x) \quad (2.24)$$

坪井らは部分的アノテーションを利用することで, ドメイン固有の表現に対応したコーパスを低コストで作成することができ, 完全なアノテーションをする場合に比べて単語分割性能を向上させることを示している.

また, Liu ら [10] は日本語と同様に単語境界のない言語である中国語の単語分割に部分的アノテーションを使用した CRF を適用し性能向上を示した. ま

<sup>§</sup><https://www.cis.upenn.edu/~treebank/>

た、彼らは crfsuite [12] を部分的アノテーションを利用できるように改良した partial-crfsuite <sup>¶</sup> を公開している。

## 2.3 部分的アノテーションを用いた日本語学習者文の単語分割

本研究では、日本語学習者の書いた日本語文に最適化された単語分割を提案する。Lang-8 から抽出した学習者の文と添削文のペアから部分的アノテーションによって学習者コーパスを自動で作成し、完全にアノテーションされた一般的なコーパスと合わせて、坪井らの提案する CRF の訓練用データとして用いる。

### 2.3.1 問題設定

今回の日本語学習者の文の単語分割は、対象文の各文字に対して、単語開始文字 (B)、単語内文字 (I)、1 文字単語 (S) のいずれかをラベル付けする系列ラベリング問題として扱い、単語分割を行う。

### 2.3.2 単語分割基準

現代日本書き言葉均衡コーパス (BCCWJ)<sup>||</sup>において採用されている短単位を単語分割の基準とする。短単位は国立国語研究所が規定したものであるが、UniDic で採用されているものは多少異なる。国立国語研究所の短単位では、意志・推量の助動詞「う」「よう」を独立した語として扱うが、UniDic における短単位では、これらを活用語尾とみなす。従って「でしょ/助動詞 う/助動詞」、「食べよ/動詞 う/助動詞」のような単語の接続を UniDic の基準では「でしょう/助動詞」、「食べよう/動詞」のように 1 単語としてみなし「意志推量形」という活用として扱う。

誤りが含まれている場合は、基本的に訂正を行った正しい文を単語分割した結果にならう。例えば、単語内部に余分な文字が含まれていたり、文字の順番がおかしいもの、単語の右隣を切り出しても意味を成さない余分な文字などに関しては一つ

<sup>¶</sup><https://github.com/ExpResults/partial-crfsuite>

<sup>||</sup>[http://pj.ninjal.ac.jp/corpus\\_center/bccwj/morphology.html](http://pj.ninjal.ac.jp/corpus_center/bccwj/morphology.html)

の単語として見る。また、余分な助詞などの単語として認識できるものが含まれている場合は、切り出す。

以下に誤りを含む文の単語分割、系列ラベリングの例を示す。「思うい」のように「思う」に「い」のような意味を成さない無駄な文字が含まれている場合は切り出さずに1単語として取り扱い単語分割を行う。

入力文 上手 じゃ ない と 思うい ます

ラベル B I B I B I S B I I B I

また、以下のように余分だが単体で意味を成す文字「が」のような単語があった場合は、前の文字「誰」から切り出す。

入力文 H G は 誰 が です か ?

ラベル B I S S S B I S S

## 2.4 実験

日本語学習者文、添削文のペアから部分的アノテーションを行って学習者コーパスを作成し、一般的なコーパスとともに学習者に特有な単語分割を部分的アノテーションを用いた CRF を用いて訓練を行う。また、訓練時に使用する学習者コーパス中の文を添削前後の挿入、削除数で制限を行い、各評価値、出力結果を確認、考察を行う。

### 2.4.1 ベースライン

提案手法との比較のためのベースラインとして、点推定を用いた単語分割器 KyTea-0.4.7 を利用する。今回 KyTea を利用する理由としては、CRF 同様に辞書を必要としない単語分割が可能であり、藤野らもベースラインとして利用しているためである。使用するモデルは、KyTea に標準で付属し BCCWJ, UniDic を主に用いて構築された**デフォルトモデル**、共に配布されている BCCWJ, UniDic 等を用いて構築された**高性能 SVM モデル**\*\*、デフォルトモデルと同様のモデルを再学

---

\*\*<http://www.phontron.com/kytea/download/model/jp-0.4.7-1.mod.gz>



習できる素性ファイル kytea-0.4.2.feat<sup>††</sup>と共にベースライン部分的アノテーションの学習者コーパスによって学習した**追加学習モデル**、BCCWJ のみで訓練を行った **BCCWJ モデル**、BCCWJ と学習者コーパスによって訓練した**再学習モデル**である。KyTea の設定はデフォルトの L2 正則化された SVM を利用し、窓幅を 3、文字・文字種 n-gram の上限を 3 とし訓練を行った。学習者コーパスを利用する際には、部分的アノテーションのされた訓練データを利用できるオプションを利用した。

配布されているモデル、素性ファイルで学習できるモデルは KyTea 特有の超短単位の単語分割のため、「語尾」のタグ付けがされた単語は前の単語に接続する。具体的には以下のようになる。

**超短単位**    ご飯/名詞   を/助詞   食べ/動詞   る/語尾

**語尾接続後**   ご飯/名詞   を/助詞   食べる/動詞

また、比較のために辞書として unidic-mecab-2.1.2<sup>‡‡</sup>を使った MeCab-0.996<sup>§§</sup>も利用した。

#### 2.4.2 データセット

水本ら [21] によって作成された言語学習者の相互添削型 SNS 「Lang-8」から抽出された学習者の文と添削文がペアになった添削コーパス<sup>¶¶</sup>を用いた。学習者文、添削文共にコメントなどがカッコ中に含まれたり、単語間にスペースが多くあったため、カッコ表現、スペースは除去を行った。そのような処理を行った日本語学習者文と添削文対 1,271,065 文を利用した。この内 500 文をランダムに選択しテストデータとして、残りを訓練用データとした。

##### 訓練用データ

訓練用データ中の日本語学習者の文、添削文のペアを利用して部分的アノテーションを行い、学習者の文に対応した単語分割の部分的アノテーション済み訓練用

---

<sup>††</sup><http://www.phontron.com/kytea/download/kytea-0.4.2.feat.gz>

<sup>‡‡</sup><https://osdn.jp/projects/unidic/releases/58338>

<sup>§§</sup><http://taku910.github.io/mecab/>

<sup>¶¶</sup><http://cl.naist.jp/nldata/lang-8/>

表 2.1 各データセットの文数

データセット		文数
現代日本語書き言葉均衡コーパス (BCCWJ)		59,431
学習者 コーパス	ins1del0 (挿入 1 以下挿入 0)	18,181
	ins2del0 (挿入 2 以下挿入 0 以下)	25,190
	ins5del5 (挿入 5 以下挿入 5 以下)	611,405
	ins5del5sub3 (挿入 5 以下挿入 5 以下)	589,058

データとなる学習者コーパスを作成する。学習者文と添削文のペア間で編集距離（削除，挿入数）が大きいものは適切な単語分割のアノテーションとならない傾向があるため，CRF の学習に使用する学習者コーパスを文対での削除，挿入数によって使用文数の制限を行った。藤野らは挿入数と削除数が 5 以下のパターンと，さらに挿入数，削除数の差分が 3 以下なパターンを学習者コーパスの利用制限としていた。今回はさらにいくつかの挿入数，削除数の制限を加えて実験を行った。

加えて，「現代日本語書き言葉均衡コーパス」(BCCWJ) のコアデータに短単位基準で単語分割された 59,431 文を，全ての箇所にも単語分割のラベル付け（フルアノテーション）済みの訓練用データとして利用する。

表 2.1 に利用するデータセットの各文数を示す。学習者コーパス中の文対における挿入数  $N$  以下，削除数  $M$  以下で利用する文の制限を行ったデータセットを  $\text{ins}N\text{del}M$  の様に表現する。加えて，挿入数，削除数の差が  $P$  以下の場合のものを  $\text{ins}N\text{del}M\text{sub}P$  のように表現する。また  $N = M = 0$  となるような文対において添削前後で変化のないものは除去した。

## テストデータ

2.3.2 項で説明した短単位基準に従い Lang-8 から抽出した日本語文 500 件に 2 人で単語分割のアノテーションを行った。片方のアノテーションを正解，もう片方のアノテーションをシステム出力と考えて一致率を評価した場合の F 値は 97.23% となった。また，単語分割のアノテーションに差異のあった文のうち，24.5% の文が誤り箇所由来する差異であった。



### 2.4.3 素性テンプレート

CRF の学習に際して、着目する文字の前 3 文字、後ろ 2 文字を着目する窓として、文字 1, 2, 3-gram, 文字種 1, 2, 3-gram を素性として用いる。具体的には以下のように文が入力された場合に、“ソ”に着目しラベル付けを行うとすると下記のように素性が選択される。

入力文 中 国 で サ ソ リ を 食 べ る .

文字 1, 2, 3-gram

1-gram : “国”, “で”, “サ”, “ソ”, “リ”, “を”

2-gram : “国で”, “でサ”, “サソ”, “ソリ”, “リを”

3-gram : “国でサ”, “でサソ”, “サソリ”, “ソリを”

文字種 1, 2, 3-gram

1-gram : “漢字”, “ひらがな”, “カタカナ”, ...

2-gram : “漢字/ひらがな”, “ひらがな/カタカナ”, “カタカナ/カタカナ”,

...

3-gram : “漢字/ひらがな/カタカナ”, “ひらがな/カタカナ/カタカナ”, ...

### 2.4.4 評価手法

単語分割の評価手法として、conlleval.pl \*\*\*の評価スクリプトを利用して、システムから出力された単語分割結果に対して、適合率、再現率、F 値を導出する。正解文に含まれる総単語数を  $N_{REF}$ 、システムの単語分割の結果に含まれる総単語数を  $N_{SYS}$ 、システムの出力のうち正解文中の単語と一致するものを  $N_{COR}$  とすると、適合率は  $N_{COR}/N_{SYS}$ 、再現率は  $N_{COR}/N_{REF}$  と定義される。また、F 値は適合率と再現率の調和平均であり、適合率を  $P$ 、再現率を  $R$  とすると、 $2 \times P \times R / (P + R)$  と定義される。各評価値の算出の具体例を示す。

正解コーパス でも じよず じゃ り ませ ん

単語分割結果 でも じよ ず じゃ り ませ ん

上記の例文の場合は、正解文の単語数が  $N_{REF} = 6$ 、システムの単語分割結果の単

\*\*\*<http://www.cnts.ua.ac.be/conll2000/chunking/output.html>

語数が  $N_{SYS} = 6$ , 分割が成功している単語は「でも」, 「ませ」, 「ん」のため  $N_{COR} = 3$  となり, 適合率は  $N_{COR}/N_{SYS} = 3/6 = 1/2$  となり, 適合率は  $N_{COR}/N_{REF} = 3/6 = 1/2$ , F 値は  $2 \times P \times R \times (P + R) = 2 \times (1/2) \times (1/2) \times (1/2 + 1/2) = 1/2$  となる.

#### 2.4.5 ツール

添削文の単語分割結果を学習者文に反映する際に **jpair** <sup>†††</sup> を利用し, 添削前後で変化のあった部分の単語分割のみアノテーションを自動で行う. また, 訓練, 結果の出力に関しては部分的アノテーションを用いた CRF の実装である **partial-crfsuite** を利用した.

#### 2.4.6 実験結果

表 2.2 に今回の実験結果の各評価値を示す. また, 以下各手法, 使用する訓練用データの組み合わせを **手法 (訓練用データ)** のように示す. また, 手法に関してはそれぞれ, **partial-crfsuite** を利用した提案手法を **P-CRF**, **KyTea-0.4.7** を利用したものを **KyTea**, **MeCab-0.996** を利用したものを **MeCab** のように表す.

P-CRF (BCCWJ) に比べ, 文対で挿入のみ行なわれた学習者コーパスを用いた P-CRF (BCCWJ+ins1del0), P-CRF (BCCWJ+ins2del0) は各評価値が向上した. しかし, 挿入数を 1 から 2 にすることで評価値が低下する. 一方, 文対で削除のみ行なわれた学習者コーパスを用いた P-CRF (BCCWJ+ins0del1) に関しては, 評価値が低下した.

KyTea の各種モデルの評価値を確認する. P-CRF と同様に BCCWJ, ins1del0 を用いて学習したモデルである KyTea (BCCWJ+ins1del0) は, BCCWJ のみで学習した KyTea (BCCWJ) に比べ各評価値が低下した. また, KyTea (BCCWJ) は適合率に関しては最も高い値となった. 素性ファイルと各種学習者コーパスを用いて学習を行った追加学習モデルはデフォルトモデル, 高性能 SVM モデルに比べ比較的悪い結果になった.

---

<sup>†††</sup><https://github.com/tkyf/jpair>

表 2.2 日本語学習者文の単語分割における各手法および訓練データの比較

手法	訓練用データ	追加文数	P(%)	R(%)	F(%)
P-CRF	BCCWJ のみ		95.67	97.12	96.39
	BCCWJ + ins1del0	18,181	<u>97.31</u>	<u>97.65</u>	<u>97.48</u>
		15,000	97.29	97.57	97.43
		10,000	97.26	97.50	97.38
	BCCWJ + ins2del0	25,190	96.90	97.42	97.16
		15,000	97.20	97.49	97.34
		10,000	97.26	97.53	97.39
	BCCWJ + ins0del1	44,227	94.51	90.59	92.51
		15,000	96.28	94.45	95.36
		10,000	96.00	93.75	94.86
KyTea	BCCWJ のみ		<b>97.44</b>	97.38	97.41
	BCCWJ + ins1del0		97.05	97.29	97.17
	デフォルト (BCCWJ+Unidic)		96.43	96.89	96.66
	高性能 SVM (BCCWJ+Unidic)		96.54	97.04	96.79
	素性ファイルのみ		90.34	91.26	90.80
	素性ファイル + ins1del0	18,181	90.40	91.32	90.86
	素性ファイル + ins2del0	25,190	90.42	91.37	90.89
	素性ファイル + ins5del5	611,405	80.52	68.04	73.75
	素性ファイル + ins5del5sub3	589,058	81.12	69.13	74.64
MeCab	BCCWJ+UniDic		97.09	<b>98.16</b>	<b>97.62</b>

同じ訓練用データを利用した KyTea (BCCWJ+ins1del0) と P-CRF (BCCWJ+ins1del0) を比較すると P-CRF を利用した場合の方が KyTea より高い評価値が得られた。



表 2.3 P-CRF (BCCWJ), P-CRF (BCCWJ+ins1del0) の削除誤り箇所の  
単語分割改善例

BCCWJ	BCCWJ+ins1del0
ダフト   パンク   は   人気   が   <u>あた</u>   と   思い   まし   た   。	ダフト   パンク   は   人気   が   <u>あ   た</u>   と   思い   まし   た   。
おもしろい   ブログ   を   書き   たい い   です   が   なに   も   <u>がんがえら</u>   ない   。	おもしろい   ブログ   を   書き   たい   です   が   なに   も   <u>がんがえ   ら</u>   ない   。

## 2.5 考察

具体的に、各モデル間で単語分割が改善、悪化した具体例を踏まえて考察を行う。また、単語分割が失敗するパターンは単語境界ではない箇所で分割してしまう**過分割**と、本来の単語境界で分割されない**未分割**に分けられる。具体的には以下の例のようなパターンである。

**過分割** けんどう | と | から | て | を | し | ます

**未分割** 待つ | て | ほが いい | です | ね | ?

また、上記の過分割、未分割を組み合わせることでその他の誤りパターンも記述できる。

### 2.5.1 挿入数、削除数の影響

P-CRF (BCCWJ) と P-CRF (BCCWJ+ins1del0) のモデル間で改善した具体例を表 2.3 に示す。学習者が文字を削除し、添削で挿入操作が行なわれる必要のある箇所の単語分割が改善された。また表 2.4 のように P-CRF (BCCWJ) は、半角英字、記号を単語分割してしまう傾向が見られたが、過分割が改善された。表 2.5 には単語分割結果が悪化した例を示す。「してる」を「して | いる」の誤りとして扱い「して | る」のように単語分割された例がいくつか見られた。実際、書き言葉では「して | いる」が正しいとされるので、学習者文中の「てる」のようなフレーズに対して添削文中「て | いる」のように添削が行われている例が表 2.6 のように

表 2.4 P-CRF (BCCWJ), P-CRF (BCCWJ+ins1del0) の半角文字箇所の  
単語分割改善例

P-CRF (BCCWJ)	P-CRF (BCCWJ+ins1del0)
で   も   、   字 幕   が   、   ”   <u>I   s   h   e   d   e a   d</u>   ?   "   と   言 っ   た   。	で   も   、   字 幕   が   、   ”   <u>Ishede</u> <u>ad</u>   ?   "   と   言 っ   た   。
H I T T   さ ん   は   日 本   に   盛 ん   (   <u>p   o   p   u   l a   r</u>   )   で す   か   ？	H I T T   さ ん   は   日 本   に   盛 ん   (   <u>popular</u>   )   で す   か   ？

表 2.5 P-CRF (BCCWJ), P-CRF (BCCWJ+ins1del0) 間の単語分割悪化例

P-CRF (BCCWJ)	P-CRF (BCCWJ+ins1del0)
今   ,   東 京   で   一 人 暮 ら し   し   <u>て る</u>   。	今   ,   東 京   で   一 人 暮 ら し   し   <u>て る</u>   。
ま も り   す ず け   て る	ま も り   す ず け   <u>て る</u>

表 2.6 ins1del0 の学習者文と添削文のペア

学習者文	添削文
オハイオ大学に行くかどうかまだ考 <u>えてる</u> 。	オハイオ大学に行くかどうかまだ考 <u>えている</u> 。
なぜあこがれ <u>てる</u> のか？	なぜあこがれ <u>ている</u> のか？

学習者コーパス中に多く確認できた。一方、テストデータの正解は UniDic を参考に「し | てる」のような単語分割で行ったため、これは不正解の事例とされた。しかしながら、学習者の単語分割として行うことを考えると、今回の提案手法でなされる単語分割の方が適切ではないだろうか。

次に、P-CRF (BCCWJ+ins1del0) から P-CRF (BCCWJ+ins2del0) に挿入数を増やした場合に表 2.7 に示すような箇所の単語分割が悪化した。「思い」のように



表 2.7 P-CRF (BCCWJ+ins1del0), P-CRF (BCCWJ+ins2del0) 間の単語分割悪化例

P-CRF (BCCWJ+ins1del0)	P-CRF (BCCWJ+ins2del0)
ダフト   パンク   は   人気   が   あ   た   と   <u>思</u>   い   まし   た   。	ダフト   パンク   は   人気   が   あ   た   と   <u>思</u>   <u>い</u>   まし   た   。
CO   は   部屋   を   買い   に   近い   所   で   探そう   の   で   、   彼女   の   代わり   今日   は   私   会社   を   当番   に   <u>行</u>   き   まし   た   。	CO   は   部屋   を   買い   に   近い   所   で   探そう   の   で   、   彼女   の   代わり   今日   は   私   会社   を   当番   に   <u>行</u>   <u>き</u>   まし   た   。
できる   こと   なら   、   あの   頃   に   戻っ   て   、   人生   を   やり直 し   て   、   夢   を   <u>叶</u>   い   ます   か   。	できる   こと   なら   、   あの   頃   に   戻っ   て   、   人生   を   やり直 し   て   、   夢   を   <u>叶</u>   <u>い</u>   ます   か   。

表 2.8 P-CRF (BCCWJ), P-CRF (BCCWJ+ins0del1) 間の単語分割悪化例

P-CRF (BCCWJ)	P-CRF (BCCWJ+ins0del1)
で   も   、   <u>付き合う</u>   の   時間   が   長い   に   なっ   たら   、	で   も   、   <u>付き合う</u>   の   時間   が   長い   に   なっ   たら   、
<u>明日</u>   は   、   ベルリン   に   、   3   8   度   でしょう   。	<u>明日</u>   は   、   ベルリン   に   、   3   8   度   でしょう   。
ここ   は   <u>寒い</u>   <u>だ</u>   から   ジョギ ング   でき   ない   。	ここ   は   <u>寒い</u>   <u>だ</u>   から   ジョギ ング   でき   ない   。

単語境界に関して左側が漢字、右側がひらがなとなるような単語をより途中で分割してしまう傾向が見られた。挿入数を 1 から 2 に増やすと、学習者コーパス中の利用する文数は 1.39 倍になったのに対して、各漢字の右側の境界で単語分割のされるアノテーション数は 1.82 倍に増加し、そのような点に影響されたのではないかと考えられる。また、改善された文は 3 文のみで、改善点はほぼ確認できなかった。文対で削除操作が行われた文を含む学習者コーパスを使った P-CRF (BC-

表 2.9 ins0del1 中の学習者文と添削文のペア

学習者文	添削文
現在の世界は以前 <u>の</u> より、怖くなる一方です。	現在の世界は以前より、怖くなる一方です。
まあ、すぐに終われるものじゃない <u>だ</u> から、これからもゆっくり楽しみましょう。	まあ、すぐに終われるものじゃないから、これからもゆっくり楽しみましょう。
恐らく来週 <u>に</u> アメリカ帰国する	恐らく来週アメリカ帰国する

表 2.10 KyTea (BCCWJ+ins1del0), P-CRF (BCCWJ+ins1del0) 間の単語分割改善例

KyTea (BCCWJ+ins1del0)	P-CRF (BCCWJ+ins1del0)
ダフト   パンク   は   人気   が   <u>あた</u>   と   思い   まし   た   。	ダフト   パンク   は   人気   が   <u>あ   た</u>   と   思い   まし   た   。
待っ   て   <u>ほが</u>   いい   です   ね   ?	待っ   て   <u>ほ   が</u>   いい   です   ね   ?

CWJ+ins0del1) は、表 2.8 に示すような箇所の単語分割が悪化した。「の」、「は」のような 1 文字の助詞、助動詞を前の単語に接続してしまうような単語分割の傾向が多く見られた。これは、日本語学習者が余分にそのような単語を挿入しがちで、表 2.9 のように、今回の学習者コーパスの作成方法では、添削前後でそのような助詞等を前の単語と接続してしまうようなラベル付けがされてしまうためである。

## 2.5.2 KyTea との比較

同じ訓練用データを利用した KyTea (BCCWJ+ins1del0) と P-CRF (BCCWJ+ins1del0) のモデル間で出力結果を比較すると表 2.10 に示すような箇所が改善された。特に、削除誤りが起こり、添削によって挿入が行なわれる必要のある箇所が良い結果となった。以上のようなことから、KyTea を利用する場合に比

表 2.11 KyTea (高性能 SVM モデル), P-CRF (BCCWJ+ins1del0) 間の単語分割改善例

KyTea (高性能 SVM モデル)	P-CRF (BCCWJ+ins1del0)
毎日   毎日  、   日本   語   を   練習   し   ましよ   う   !	毎日   毎日  、   日本   語   を   練習   し   ましょう   !
ビール   を   飲め   ば   後   課題   が   でき   ない   よう   に   なる   ん   だろ   う   と   思っ   て   …	ビール   を   飲め   ば   後   課題   が   でき   ない   よう   に   なる   ん   だろう   と   思っ   て   …

べ、部分的アノテーションを利用した CRF を利用する場合の方がより学習者コーパスの影響が良くも悪くも大きいことがわかる。

また、BCCWJ 単体で学習者文の単語分割にある程度有効であることが確認できた。BCCWJ は書籍や新聞中の整った文だけでなく、ブログやネット掲示板等のいわば整っていない文を含んでいるため、学習者文の単語分割に有用であったのではないかと考えられる。

KyTea を利用したモデルで一番評価値の高かった高性能 SVM モデル、P-CRF (BCCWJ+ins1del0) 間で P-CRF の良かった例を表 2.11 に示す。KyTea における高性能モデルでは、「でしょう」、「やろう」などの今回採用されている短単位において意志推量系と判断される活用形が「でしよ | う」、「よろ | う」のように分割されてしまう傾向が確認できた。今回、正解データの単語分割基準を 2.3.2 項で説明したものとしたため、「でしょう」、「やろう」のような単語は助動詞、動詞の意志推量形として扱われる。一方 KyTea で採用されている単語分割基準の超短単位では、「で/動詞 しよ/語尾 う/助動詞」、「や/動詞 ろ/語尾 う/助動詞」のように単語分割がされ、今回は語尾を前の単語に接続する設定で単語分割を行っているので「でしよ | う」、「よろ | う」のように分割される。その他の KyTea と共に配布されているモデル、素性ファイルを用いたモデルとの比較ではこれらと同様の違いが誤りとして出てしまった。このため、提案手法との比較には不適切であった。

今回、藤野らの実験と比較して KyTea (素性ファイル +ins5del5), KyTea (素性ファイル +ins5del5sub3) の結果が大きく性能が低下してしまった。素性ファイ



表 2.12 MeCab, P-CRF (BCCWJ+ins1del0) 間の単語分割改善例

MeCab	P-CRF (BCCWJ+ins1del0)
おもしろい   ブログ   を   書き   たい   です   が   なに   も   <u>がん</u>   が   えら   ない   。	おもしろい   ブログ   を   書き   たい   です   が   なに   も   <u>がんがえ</u>   ら   ない   。
今日   、   友達   と   一緒   に   <u>ケーキ屋</u>   を   通し   た   とき   、   後   で   ケーキ   を   買おう   と   友 達   が   言っ   た   とき   、   突然   、   今日   は   友達   の   誕生   日   と   さっぱり   思い出し   た   。	今日   、   友達   と   一緒   に   <u>ケーキ屋</u>   を   通し   た   とき   、   後   で   ケーキ   を   買おう   と   友達   が   言っ   た   とき   、   突然   、   今日   は   友達   の   誕生   日   と   さっぱり   思い出し   た   。

ルが超短単位で学習されるのに対して、学習者コーパスが短単位でアノテーションされたものであったため単語分割の基準にずれが生じうまく訓練できなかったことと、テストデータを新たに短単位でアノテーションしたことが理由として考えられる。

### 2.5.3 MeCab との比較

表 2.12, 2.13 にそれぞれ, MeCab と P-CRF (ins1del0) 間で単語分割が改善した例, 悪化した例を示す。ひらがなを多く含み, 誤りのある箇所, 変換ミスのある箇所等が改善されたことが確認できた。また, 複数の語で構成され 1 語となるような固有名詞や複合動詞などを分割しすぎたり, 連続する 1 文字で 1 単語を構成する漢字を接続する傾向が悪化点としてあげられる。MeCab との比較では, 複数の語で構成され 1 語となるような固有名詞や複合動詞などの分割がうまくいっていないことが分かったが, これは今回の手法では辞書の参照をシステム中で行っていないことに起因する。



表 2.13 MeCab, P-CRF (BCCWJ+ins1del0) 間の単語分割悪化例

MeCab	P-CRF (BCCWJ+ins1del0)
「 <u>ほうき星</u> 」の <u>譜</u> が <u>欲</u> <u>い</u> です。	「 <u>ほうき</u> <u>星</u> 」の <u>譜</u> が <u>欲</u> <u>しい</u> です。
この <u>まま</u> <u>捨て置く</u> <u>わけ</u> に は <u>ゆかぬ</u> 」と <u>仰い</u> <u>まし</u> <u>た</u> 。	この <u>まま</u> <u>捨て</u> <u>置く</u> <u>わけ</u> に は <u>ゆかぬ</u> 」と <u>仰い</u> <u>まし</u> <u>た</u> 。
十分 <u>配慮</u> <u>さ</u> <u>れ</u> <u>て</u> <u>い</u> <u>ない</u> <u>アンケート</u> <u>で</u> <u>すい</u> <u>ませ</u> <u>ん</u> 。	十 <u>分</u> <u>配慮</u> <u>さ</u> <u>れ</u> <u>て</u> <u>い</u> <u>ない</u> <u>アンケート</u> <u>ですい</u> <u>ませ</u> <u>ん</u> 。
赤 <u>信号</u> の <u>時道</u> を <u>渡っ</u> <u>て</u> は <u>いけ</u> <u>ませ</u> <u>ん</u> <u>よ</u> 。	赤 <u>信号</u> の <u>時道</u> を <u>渡っ</u> <u>て</u> <u>は</u> <u>いけ</u> <u>ませ</u> <u>ん</u> <u>よ</u> 。

## 2.6 まとめと今後の課題

部分アノテーションを利用した CRF に、BCCWJ と学習者コーパス中の添削前後で挿入数が 1 の文を学習に利用することで、BCCWJ 単体の場合に比べ評価値が向上し、誤り箇所についてもうまく単語分割ができるようになったことを確認した。しかし、学習者コーパス中の添削前後で削除が行われた文の部分的アノテーションがうまくいっていないことが確認され、今後これらのデータを有効に利用する手法の検討が必要である。加えて、MeCab と比較して複合語等の単語分割がうまくいかなかったため、システム中で辞書の参照を導入したい。

また、2.2.1 項において示した齊藤らのように、あらかじめ学習者が誤りやすいパターンについては最初から与えて学習を行うことも有効ではないかと考えられる。

## 第3章 深層ニューラルネットを利用した乱れた日本語の頑健な単語分割

### 3.1 導入

本章では、日本語同様に単語間に空白のない言語である中国語の単語分割において近年高い精度を示している深層ニューラルネットを用いた手法を用いる。中国語における多くの手法では、アノテーション済みの大量のテキストから直接文字の分散表現と深層ニューラルネットのパラメータを学習している。一方、日本語文の単語分割には漢字、ひらがな、カタカナのように様々な文字種が関係していると考えられる。したがって本手法では文字に加えて文字種を分散表現として深層ニューラルネットへの入力として利用する。

また、分野適応のために予め分野適応先のコーパスで入力に用いる文字の分散表現の学習を行うこともしくは、分野適応先のコーパスを利用して追加学習を行うことの2つの手法を提案する。実験によって、本手法における分野適応では問題があることが明らかになった。

### 3.2 関連研究

単語分割タスクは各文字に対して、その文字で単語が切れるもしくは単語が切れないなどを表すラベルをラベル付けする系列ラベリング問題として扱うことができる。Zheng ら [15], Pei ら [13], Chen ら [2, 3] は Collobert ら [5] の提案する系列ラベリング問題を深層学習によって解く手法を中国語の単語分割タスクに応用することで、既存の手法に比べて素性選択の煩わしさを解消し、性能向上を図っている。

深層学習を利用した文への極性付与などのタスクでは、文中の各単語に対応した分散表現と呼ばれるベクトルを深層ニューラルネットへの最初の入力とするが、上記の中国語の単語分割タスクにおいては各文字へラベル付けを行うため各文字に対して分散表現を定義し入力として用いる。着目する文字の任意の窓枠の文字に対しての分散表現を連結し、ニューラルネットへ入力することで各ラベルに対しての確率値を得る。

特に, Chen ら [3] は Long short-term memory ニューラルネットワークを適用することでネットワーク中のセルに前の重要な情報を保持することで窓幅の制限を克服し, 文中で隣接していない箇所の関係性を利用することで現時点での最高精度を達成している.

今回はこれらの研究を参考に, 中国語同様に単語間に空白のない日本語文の単語分割に取り組む. しかし, 日本語はひらがな, カタカナ, 漢字のように複数の文字種が存在し, 各文字が音的な情報で対応していることが中国語と異なる. そのため, 本研究では学習の際に文字だけでなく文字種も分散表現として加える. また, 今回は日本語学習者文の単語分割への分野適応を深層ニューラルネットの一種である双方向の LSTM (Bidirectional Long-Short Term Memory) ネットワークを用いて行う点が先行研究と異なる.

### 3.3 提案手法

本節では, 2 章と同様に単語分割を系列ラベリング問題と捉え, 深層ニューラルネットを利用して日本語単語分割をする手法を示し, 日本語学習者文の単語分割に分野適応するための手法を提案する.

#### 3.3.1 分散表現

ニューラルネットを利用してシンボリックなデータを扱う際に, それらを分散表現とよばれる分散的なベクトルで表現する. [1, 4]

日本語の単語分割タスクにおいて, 特に文字に対して着目した場合に大きさ  $|C|$  の辞書  $C$  を利用するとする. 各文字  $c \in C$  は実数値のベクトル (分散表現)  $v_c \in \mathbf{R}^d$  で表現され, ここで  $d$  はベクトル空間の次元数となる. そして文字分散表現は分散行列  $\mathbf{M} \in \mathbf{R}^{d \times |C|}$  に積み重ねられる. 文字  $c \in C$  に対応した文字分散表現  $v_c \in \mathbf{R}^d$  はルックアップテーブル層で取得される.

今回は, 日本語の単語分割タスクのために文字種も加えて考慮したいため, 文字種の分散表現も文字同様に定義を行う. 各文字種  $t \in T$  は分散表現  $v_t \in \mathbf{R}^e$  で表現され,  $e$  はベクトル空間の次元数とし, 分散行列は  $\mathbf{N} \in \mathbf{R}^{e \times |T|}$  となり, 文字種



$t \in T$  に対応した文字種分散表現  $v_t \in \mathbf{R}^e$  は文字の場合と同様にルックアップテーブル層で取得される。

また、日本語学習者文の分野適応のために word2vec の手法を利用し、分野適応先の生データを用いて文字の分散表現の事前学習を行う手法も利用した。以下 char2vec と表現する。

### 3.3.2 ニューラルモデルを利用した日本語単語分割

ニューラルモデルを利用した単語分割モデルは大きく 3 つのレイヤーに分けられる: (1) 文字, 文字種分散表現層; (2) ニューラルネットワーク層; (3) ラベル推定層。

本研究においては、文字のラベル付けは隣接する文字に依存していることを仮定し、2 番目のニューラルネットワーク層に対して、双方向の LSTM を利用した Bidirectional-LSTM [8] を採用し、全体図を 3.1 に示す。加えて、各 LSTM ユニットの図 3.2 のような構造を持つ。

LSTM ユニットのステップ毎に入力が観測される度に記憶を更新するメモリーセルを持つ。セルは入力ゲート  $i$ 、忘却ゲート  $f$ 、出力ゲート  $o$  の三つのゲートで制御される。ゲートにおける演算は、要素ごとの積で定義され、ゲートが 0 でない場合は入力値がスケールされ、0 の場合は入力が除外される。出力ゲートの出力は、時刻  $t$  時にニューラルネットにおける次の層の入力となり、次の時刻  $t+1$  の際の過去の隠れ状態として埋め込まれる。各ゲートの定義、セルのアップデート、出力については以下のようなになる。ここでは、 $\sigma$  は sigmoid 関数、 $\phi$  は tanh 関数を意味する。

$$\mathbf{i}_t = \sigma(\mathbf{W}_{ix}\mathbf{x}_t + \mathbf{W}_{ih}\mathbf{h}_{t-1} + \mathbf{W}_{ic}\mathbf{c}_{t-1}) \quad (3.31)$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{fx}\mathbf{x}_t + \mathbf{W}_{fh}\mathbf{h}_{t-1} + \mathbf{W}_{fc}\mathbf{c}_{t-1}) \quad (3.32)$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \phi(\mathbf{W}_{cx}\mathbf{x}_t + \mathbf{W}_{ch}\mathbf{h}_{t-1}) \quad (3.33)$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{ox}\mathbf{x}_t + \mathbf{W}_{oh}\mathbf{h}_{t-1} + \mathbf{W}_{oc}\mathbf{c}_t) \quad (3.34)$$



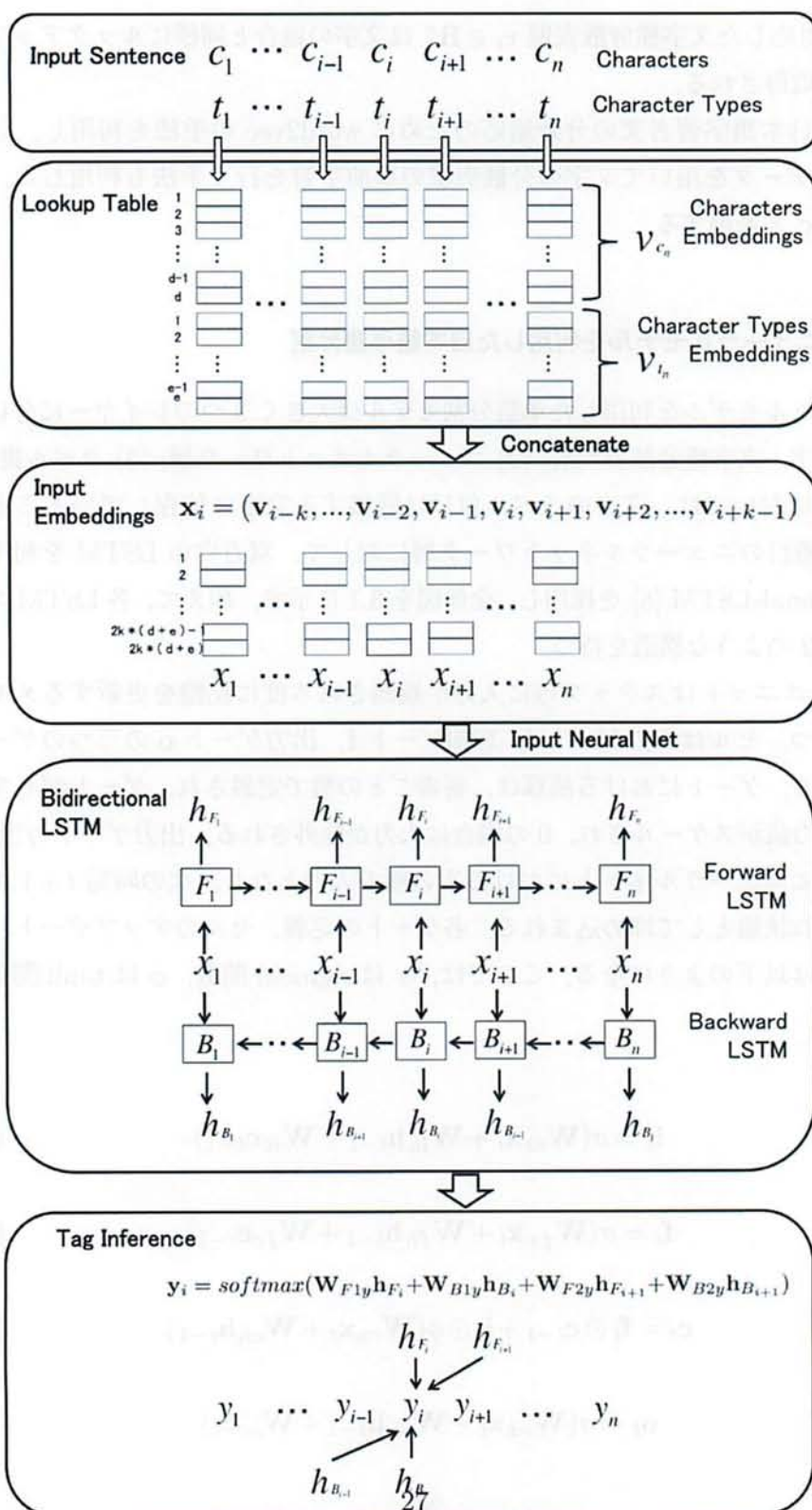


図 3.1 ニューラル単語分割モデル

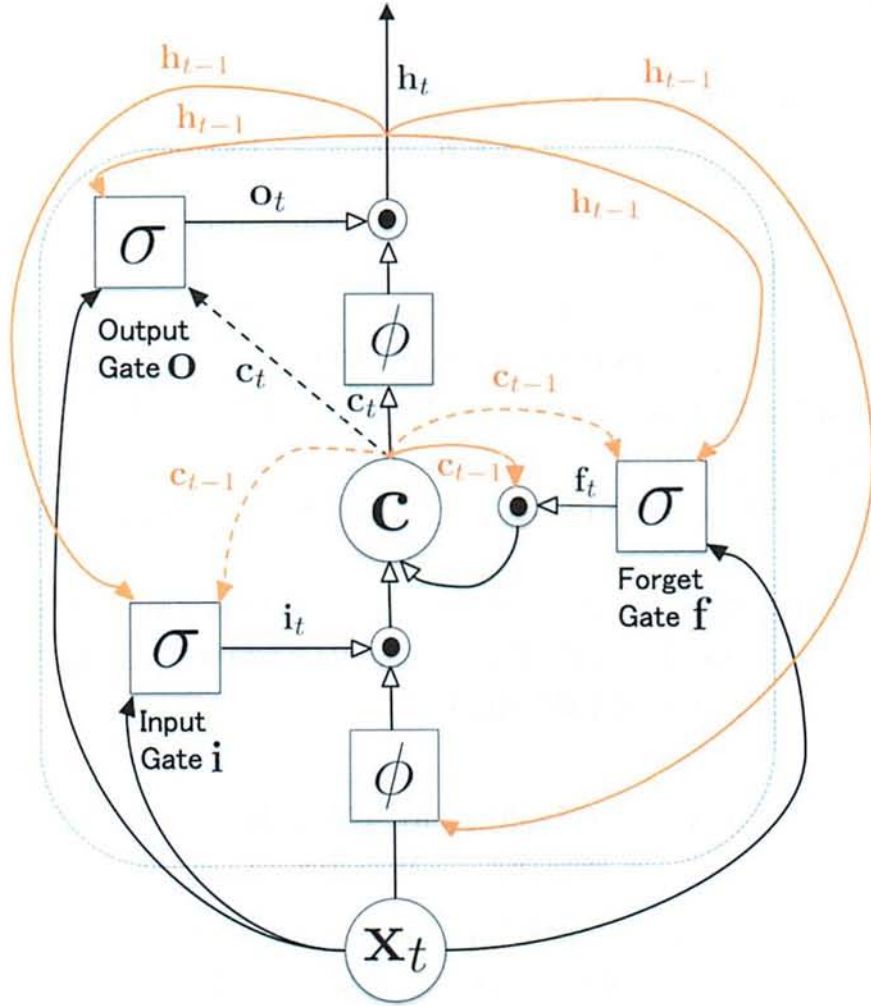


図 3.2 LSTM メモリーユニット

$$\mathbf{h}_t = \mathbf{o}_t \odot \phi(\mathbf{c}_t) \quad (3.35)$$

始めに、長さ  $n$  の入力文  $c_{(1:n)} = (c_1, c_2, \dots, c_i, \dots, c_n)$  が与えられた時、文中の  $i$  番目の文字  $c_i$ 、文字種  $t_i$  に対して、ルックアップテーブルから参照された文字、文字種の分散表現をそれぞれ  $v_{c_i}, v_{t_i}$  とし、それらを連結したベクトルを  $\mathbf{v}_i = (v_{c_i}, v_{t_i})$  とする。また、前後の文字を考慮するため、窓幅  $k$  とすると、 $i$  番目の文字  $c_i$  に対してのニューラルネットへの入力となる文脈ベクトル  $\mathbf{x}_i$  は前後の文字、文字種の

分散表現を連結したものとなり以下の式で表現できる.

$$\begin{aligned} \mathbf{x}_i &= (\mathbf{v}_{i-k}, \dots, \mathbf{v}_{i-2}, \mathbf{v}_{i-1}, \mathbf{v}_i, \mathbf{v}_{i+1}, \mathbf{v}_{i+2}, \dots, \mathbf{v}_{i+k-1}) \\ \mathbf{x}_i &\in \mathbf{R}^{H_1}, H_1 = 2 \times k \times (d+e) \end{aligned} \quad (3.36)$$

従って入力文  $s_{(1:n)}$  に対応したベクトル列  $\mathbf{x}_{(1:n)} = (\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_i, \dots, \mathbf{x}_n)$  がニューラルネット層の入力となる.

ニューラルネット層に入力された, ベクトル列  $\mathbf{x}_{(1:n)}$  中の各ベクトル  $\mathbf{x}_i$  は図 3.1 に示すように前向きの LSTM 層 (Forward LSTM)  $F_n$  と, 後ろ向きの LSTM 層 (Backward-LSTM)  $B_n$  に入力される. Forward-LSTM は  $F_1, F_2, \dots, F_{n-1}, F_n$  の順に状態遷移をするのに対して, Backward-LSTM は逆順の  $F_n, F_{n-1}, \dots, F_2, F_1$  で状態遷移が行われる. 各 LSTM ユニットから隠れ層ベクトル  $h_{F_i}, h_{B_i}$  が出力される.

最終的に以下の式から各文字  $c_i$  に対してのラベル付けの確率値の分布  $\mathbf{y}_i \in \mathbf{R}^L$  が与えられる. この時,  $L$  は文字に対してラベル付けの種類数であり,  $\mathbf{y}_i$  の各要素の値  $y_i^l$  はラベル  $l$  をつける確率値に対応する.

$$\begin{aligned} \mathbf{y}_i &= \text{softmax}(\mathbf{W}_{F1y} \mathbf{h}_{F_i} + \mathbf{W}_{B1y} \mathbf{h}_{B_i} \\ &\quad + \mathbf{W}_{F2y} \mathbf{h}_{F_{i+1}} + \mathbf{W}_{B2y} \mathbf{h}_{B_{i-1}}) \end{aligned} \quad (3.37)$$

従って, 入力文  $c_{(1:n)}$  に対して最も尤もらしいラベル系列  $\hat{y}_{(1:n)} = (\hat{y}_1, \hat{y}_2, \dots, \hat{y}_i, \dots, \hat{y}_n)$  は, 各文字  $c_i$  に対して最も確率の高いラベル  $\hat{y}_i$  を以下の式で求めることにより定まる.

$$\hat{y}_i = \arg \max_l y_i^l \quad (3.38)$$

### 3.3.3 訓練

モデルの訓練のために, 交差エントロピー誤差を利用する. 与えられた文  $c_{(1:n)}$  に対して, 正しいラベル系列を  $y_{(1:n)}$  とする. モデルのパラメータセットを  $\theta$  とす

ると、予測されたラベル系列  $\hat{y}_{1:n}$  に対して定義される交差エントロピー誤差は次のようになる。また、最適化手法には AdaGrad [7] を利用した。

加えて、分野適応のために 2.2.3 節で説明した部分的にアノテーションのされた学習者コーパスを利用し、追加学習する際には曖昧なラベル付けのされた文字の誤差は無視して訓練を行う。

$$J(\theta) = - \sum_{i=1}^n \sum_{l=1}^{|T|} \delta_{l\hat{y}_i} \log y_i^l \quad (3.39)$$

$$\delta_{l\hat{y}_i} = \begin{cases} 1 & (l = \hat{y}_i) \\ 0 & (l \neq \hat{y}_i) \end{cases}$$

### 3.4 実験

提案手法とベースライン間における日本語学習者の単語分割の性能を比較する。

2.4.1 で利用したベースラインと、2.4.6 で示した提案手法の P-CRF (BCCWJ), P-CRF (BCCWJ+ins1del0) を比較対象として用いる。

2.4.2 で利用した BCCWJ コーパス、添削コーパスから作成した訓練用データをモデルの訓練に利用する。文字分散表現の事前学習にはアノテーションのされていない添削コーパスの訓練用データ中の全日本語学習者文を用いる。追加学習 (retrain) のためには、部分的アノテーションがされ、削除、挿入数で使用文数を制限した ins1del0 (挿入 1 以下挿入 0) の学習者コーパスを利用する。また、テストデータは 2.4.2 で利用した日本語学習者文 500 件とする。

#### 3.4.1 ツール, パラメータセット

今回の提案手法の実装には、深層学習のフレームワークである `chainer`[14]\* を利用した。また、モデルのパラメータは表 3.1 に示す。

---

\*<http://chainer.org>



表 3.1 パラメータ

辞書全文字数	$ C  = 2500$
文字分散表現次元	$d = 100$
全文字種数	$T = 8$
文字種分散表現次元	$e = 8$
隠れ層次元	$H_2 = 100$
窓幅	$k = 3$
初期学習率	$\alpha = 0.01$
正則化	$\lambda = 0$
エポック数	$E = 20$

### 3.4.2 実験結果

表 3.2 に今回の実験結果の各評価値を示す。また、以下各手法、使用する訓練用データの組み合わせを **手法 (訓練用データ)** のように示す。また、手法に関してはそれぞれ、本提案手法における Bidirectional-LSTM を利用したものを **B-LSTM**、加えて、文字分散表現を利用した提案手法を **B-LSTM (訓練用データ (char2vec))**、部分的アノテーションのされた訓練データで再学習されたものを **B-LSTM (訓練用データ (retrain))**、2 章で提案した手法を **P-CRF**、KyTea-0.4.7 を利用したものを **KyTea**、MeCab-0.996 を利用したものを **MeCab** のように表す。

今回の提案手法に関しては、各値が P-CRF と比較して良い結果を出すことができなかった。B-LSTM (BCCWJ) と比較して、再学習をすることで分野適応を行った B-LSTM (BCCWJ + ins1del0 (retrain)) は再学習時に学習率を小さい値にすることで多少精度が良くなった。また、char2vec を利用し事前学習を行い再学習を行ったものはいい結果が得られなかった。

### 3.5 考察

具体的に、本章で提案した B-LSTM を用いた単語分割の結果を、2.5 節と同様に実際の出力結果を踏まえて考察を行う。

## B-LSTM, P-CRF 間の比較

今回提案した B-LSTM を用いた手法と、P-CRF を用いた手法に関して、訓練データを BCCWJ のみにした場合の結果の比較を行う。B-LSTM (BCCWJ) の評価値は適合率 (Precision) 以外の点において P-CRF (BCCWJ) に劣る結果となってしまった。実際の単語分割結果における違いは誤り箇所、ひらがな箇所が多く見られ、表 3.3 に悪化した例を、表 3.4 に改善された例を示す。誤り箇所、ひらがな箇所では B-LSTM, P-CRF の双方で単語分割結果が異なる文が多く見られたが特徴的なものを確認できなかった。しかしながら、表 3.5 に示すようにアルファベットの単語、カタカナの複合語に対しては異なる結果が得られた。P-CRF では過分割する傾向が見られ、B-LSTM に関しては正解データに合うような分割になる傾向が見られた。

具体的になぜ上記のような、結果になるのかは不明であるが、今回の B-LSTM では、ラベル付けする文字の前後の窓幅内に位置する文字に対しての unigram を分散表現として捉え、窓幅内の分散表現を連結してニューラルネットへの入力とした。その一方で、P-CRF ではラベル付けの文字の前後の窓幅内に位置する文字に対しての 1, 2, 3-gram を入力の素性としており、情報量に差があるのではないかと考えられる。従って、B-LSTM においても、ラベル付けする文字の前後の窓幅内に位置する文字に対して 1-gram だけではなく 2,3-gram に対しての分散表現を入力として利用することで改善の可能性がある。

## B-LSTM (BCCWJ), B-LSTM (BCCWJ + Lang-8 (char2vec)) 間の比較

B-LSTM に対して、Lang-8 の日本語学習者文によって事前学習を行った文字分散表現 (char2vec) を初期値として利用することでどのような影響があったかを確認する。評価値を見ると、char2vec を利用した B-LSTM (BCCWJ + Lang-8 (char2vec)) の方が劣る結果となってしまった。実際の出力結果を確認すると、表 3.6 のようなアルファベット、カタカナ語の複合語に関して過分割になってしまう傾向が見られた。また、表 3.7 に示すように、誤りが含まれていない箇所の単語分割が悪化した。逆に、表 3.9, 3.10 に示すように、誤りを含む箇所や漢字への変換がされていないひらがな箇所の単語分割の改善が見られた。テストセット全体の評価値は下がっているが、誤りや表記揺れを含む文に対しての単語分割の改善が見られ

る一方、正規の文への単語分割結果が悪化しているため、誤りや表記揺れを含む文に対して過剰に適合しているのではないかと考えられる。従って、事前に分散表現を学習する際に Lang-8 の日本語学習者文に加えて、BCCWJ などの一般的なコーパスも用いることで改善できるのではないと思われる。

### B-LSTM (BCCWJ), B-LSTM (BCCWJ + ins1del0 (retrain)) 間の比較

BCCWJ で予め学習を行った B-LSTM に対して、部分的アノテーションを行った日本語学習者文を利用して再学習を行った B-LSTM (BCCWJ + ins1del0 (retrain)) についての結果を確認する。各評価値に関しては、再学習を行わなかった場合の B-LSTM (BCCWJ) と比較して、再学習時に学習率を  $\alpha = 0.00001$  にした場合若干結果が良くなった。また、表 3.11 のように未分割のものが改善され、表 3.12 のように正しいものが過分割のように悪化する傾向が見られた。再学習でより分割する傾向が得られたと考えられる。

## 3.6 まとめと今後の課題

深層ニューラルネットの一種である Bidirectional-LSTM を利用して日本語学習者文の単語分割へと分野適応する手法を検討した。分野適応のために、アノテーションのされていない日本語学習者文コーパスから word2vec の手法を適用し、文字分散表現を事前学習する手法と部分的に単語分割のアノテーションのされた日本語学習者文コーパスを用いて再学習する手法を提案した。しかしながら、本章における提案手法では精度の改善は確認することができなかった。特に、部分的アノテーションのされた学習者コーパスを用いた再学習を行う手法は本章で提案した、Bidirectional-LSTM を用いたニューラルネットワークのモデルに対して不適切であり、2 章において提案した CRF の拡張モデルに対してのみ有効であることがわかった。

全体の精度の改善は達成できなかったが、誤りや表記揺れの含まれている文の単語分割が改善されることを実験結果から確認することができ、今後の改善につながる知見が得られた。今後の課題として、文字分散表現の事前学習の際に日本語学習者文のコーパスに加えて一般的な BCCWJ などのコーパスを利用することと、ペー



スとなるニューラルネットワークへの入力の際に、着目する文字の窓幅の 1-gram の分散表現だけではなく、2,3-gram の分散表現も利用すること、ニューラルネットワークの最適なパラメータセットの探索などがあげられる。



表 3.2 日本語学習者文の単語分割における各手法および訓練データの比較

手法	訓練用データ	char2vec	文字種	P(%)	R(%)	F(%) (epoch)
B-LSTM (提案手法)	BCCWJ + Lang-8 (char2vec)	on	on	96.54	95.79	96.17 (19)
		on	off	95.95	95.73	95.84 (14)
	BCCWJ	off	on	96.43	96.02	96.23 (8)
		off	off	95.01	95.12	95.07 (8)
	BCCWJ + ins1del0 (retrain)	off	on	79.82	82.74	81.25 (1)
	BCCWJ + ins1del0 (retrain 学習率 $\alpha = 0.00001$ )	off	on	96.33	96.16	96.25 (9)
	BCCWJ + Lang-8 (char2vec) + ins1del0 (retrain 学習率 $\alpha = 0.00001$ )	on	on	96.54	95.79	96.17 (1)
P-CRF	BCCWJ のみ			95.67	97.12	96.39
	BCCWJ + ins1del0			<u>97.31</u>	<u>97.65</u>	<u>97.48</u>
KyTea	BCCWJ のみ			97.44	97.38	97.41
	BCCWJ + ins1del0			97.05	97.29	97.17
	デフォルト (BC-CWJ+Unidic)			6.43	96.89	96.66
MeCab	BCCWJ+UniDic			97.09	98.16	97.62

表 3.3 P-CRF (BCCWJ), B-LSTM (BCCWJ) の単語分割悪化例 - 誤り箇所, ひらがな箇所

P-CRF (BCCWJ)	B-LSTM (BCCWJ)
しごと   の   よう   、   <u>新しい</u>   くつ   が   か   い   つ   た   。	しごと   の   よう   、   <u>新しいく</u>   つ   が   か   い   つ   た   。
<u>かぞく</u>   は   ぜんぜん   うるさい   です   よ   。	<u>かぞく</u> は   ぜんぜん   うるさい   です   よ   。

表 3.4 P-CRF (BCCWJ), B-LSTM (BCCWJ) の単語分割改善例 - 誤り箇所, ひらがな箇所

P-CRF (BCCWJ)	B-LSTM (BCCWJ)
生まれ   て   から   ずっと   同じ   町   に   住ん   で   い   た   の   で   友   だち   会え   なく   て   <u>さび</u>   し   かつ   た   です   。	生まれ   て   から   ずっと   同じ   町   に   住ん   で   い   た   の   で   友   だち   会え   なく   て   <u>さびし</u>   かつ   た   です   。
と   で   も   尊敬   です	と   で   も   尊敬   です

表 3.5 P-CRF (BCCWJ), B-LSTM (BCCWJ) の単語分割改善例 - アルファベット, カタカナ語の複合語

P-CRF (BCCWJ)	B-LSTM (BCCWJ)
研究   内容   の   <u>P</u>   <u>PT</u>   を   作っ   て   自分   の   パソコン   を   持っ   て   も   よろしい   でしょう   か   。	研究   内容   の   <u>PPT</u>   を   作っ   て   自分   の   パソコン   を   持っ   て   も   よろしい   でしょう   か   。
実   は   この   <u>ウェブ</u>   <u>サイト</u>   を   見   つけ   た   の   は   偶然   の   こと   だ   。	実   は   この   <u>ウェブサイト</u>   を   見   つけ   た   の   は   偶然   の   こと   だ   。

表 3.6 B-LSTM (BCCWJ), B-LSTM (BCCWJ+Lang-8 (char2vec)) の単語分割悪化例 - アルファベット, カタカナ

B-LSTM (BCCWJ)	B-LSTM (BCCWJ+Lang-8 (char2vec))
で   から   、   私   たち   は   テ-ケアウエイ   を   かつ   て   、   パブ   で   たべ   まし   た   。	で   から   、   私   たち   は   テ-ケ   ア   ウエイ   を   かつ   て   、   パブ   で   たべ   まし   た   。
年   の   と   き   LastFriends   と   い う   ドラマ   の   思   い   を   掛   け   ま し   て   、   い   ろ   い   ろ   考   え   まし た   。	年   の   と   き   L   astFrien   ds   と   い   う   ドラマ   の   思   い   を   掛   け   まし   て   、   い   ろ   い   ろ   考   え   ま し   た   。

表 3.7 B-LSTM (BCCWJ), B-LSTM (BCCWJ+Lang-8 (char2vec)) の単語分割悪化例 - 誤りが含まれていない箇所

B-LSTM (BCCWJ)	B-LSTM (BCCWJ+Lang-8 (char2vec))
CO   は   部屋   を   買   い   に   近   い   所   で   探   そ   う   の   で   、   彼   女   の   代   わ   り   今   日   は   私   会   社   を   当   番   に   行   き   まし   た   。	CO   は   部屋   を   買   い   に   近   い   所   で   探   そ   う   の   で   、   彼   女   の   代   わ   り   今   日   は   私   会   社   を   当   番   に   行   き   まし   た   。
こ   ん   な   厚   恩   は   課   長   に   も ら   わ   不   い   は   ず   だ   ろ   う   、   と   く に   、   僕   は   前   に   と   て   も   失 礼   だ   っ   た   の   に   。	こ   ん   な   厚   恩   は   課   長   に   も ら   わ   不   い   は   ず   だ   ろ   う   、   と   く に   、   僕   は   前   に   と   て   も   失 礼   だ   っ   た   の   に   。

表 3.8 B-LSTM (BCCWJ), B-LSTM (BCCWJ+Lang-8 (char2vec)) の単語分割悪化例 - 誤りを含む箇所

B-LSTM (BCCWJ)	B-LSTM (BCCWJ+Lang-8 (char2vec))
今日   私   の   <u>同りよう</u>   は   彼   の   お   母   さん   が   作っ   て   お   菓子   に   くれ   ます   。	今日   私   の   <u>同   りよう</u>   は   彼   の   お   母   さん   が   作っ   て   お   菓子   に   くれ   ます   。
私   が   ひつよう   と   なる   <u>かんけい</u>   。	私   が   ひつよう   と   なる   <u>か   ん   けい</u>   。

表 3.9 B-LSTM (BCCWJ), B-LSTM (BCCWJ+Lang-8 (char2vec)) の単語分割改善例 - 誤りを含む箇所

B-LSTM (BCCWJ)	B-LSTM (BCCWJ+Lang-8 (char2vec))
お   つまみ   と   し   て   、   蜂   の   子   は   美味しい   と   聞き   まし   た   の   で   、   食べ   て   <u>みようか</u>   と   、   その   時   に   考え   まし   た   。	お   つまみ   と   し   て   、   蜂   の   子   は   美味しい   と   聞き   まし   た   の   で   、   食べ   て   <u>みよう   か</u>   と   、   その   時   に   考え   まし   た   。
ピンポン   と   <u>バス   ケトボル</u>   が   得意   です   。	ピンポン   と   <u>バスケットボル</u>   が   得意   です   。
そう   いえ   ば   、   この   仕事   が   <u>唯派   遣</u>   の   仕事   です   が   、   会社   員   と   変わっ   た   に   できる   か   、   でき   ない   の   か   、   わたし   も   分から   ない   な   。	そう   いえ   ば   、   この   仕事   が   <u>唯   派遣</u>   の   仕事   です   が   、   会社   員   と   変わっ   た   に   できる   か   、   でき   ない   の   か   、   わたし   も   分から   ない   な   。



表 3.10 B-LSTM (BCCWJ), B-LSTM (BCCWJ+Lang-8 (char2vec)) の  
単語分割改善例 - 漢字変換無し (ひらがな) 箇所

B-LSTM (BCCWJ)	B-LSTM (BCCWJ+Lang-8 (char2vec))
かぞくは   ぜんぜん   うるさい   です   よ  。	かぞく   は   ぜんぜん   うるさい   で す   よ  。
たぶん   ね  、  もっともわれわれ   を   離れる   よう   に   する   の   は   言葉   な   ん   でしょう  。	たぶん   ね  、  もっとも   われわれ   を   離れる   よう   に   する   の   は   言葉   な   ん   でしょう  。
で   も   一   日   <u>ずつ</u>   こんな   よう   に   泳ん   で  、  泳げ   ば   泳ぐ   ほど   水   の   世界   が   好き   で す  。	で   も   一   日   <u>ずつ</u>   こんな   よ う   に   泳ん   で  、  泳げ   ば   泳 ぐ   ほど   水   の   世界   が   好き   です  。

表 3.11 B-LSTM (BCCWJ), B-LSTM (BCCWJ+ins1del0 (retrain 学習  
率  $\alpha = 0.00001$ )) の単語分割改善例

B-LSTM (BCCWJ)	B-LSTM (BCCWJ+ins1del0 (re- train))
かぞくは   ぜんぜん   うるさい   です   よ  。	かぞく   は   ぜんぜん   うるさい   で す   よ  。
初   対面   の   際   の   服装   は   と ても   大切   な   こと   と   いっ   て   も  、  態度   や   話し   方   は   よく   なけれ   ば   いい   印象   を   <u>残られ</u>   ませ   ん  。	初   対面   の   際   の   服装   は   と ても   大切   な   こと   と   いっ   て   も  、  態度   や   話し   方   は   よく   なけれ   ば   いい   印象   を   <u>残ら</u>   れ   ませ   ん  。
「   だめ   <u>だよ</u>  、  後輩   は   こ こ   に   いる   もの  。  」   と   手   を   振っ   て  、  拒否   し   まし た  。	「   だめ   <u>だ</u>   よ  、  後輩   は   こ こ   に   いる   もの  。  」   と   手   を   振っ   て  、  拒否   し   まし た  。

表 3.12 B-LSTM (BCCWJ), B-LSTM (BCCWJ+ins1del0 (retrain 学習率  $\alpha = 0.00001$ ))) の単語分割悪化例

B-LSTM (BCCWJ)	B-LSTM (BCCWJ+ins1del0 (retrain))
五   分   後   再び   呼び   、   この   際   に   ぜひ   起き   <u>られ</u>   て   いる   。	五   分   後   再び   呼び   、   この   際   に   ぜひ   起き   <u>ら</u>   <u>れ</u>   て   いる   。
生まれ   て   から   ずっと   同じ   町   に   住ん   で   い   た   の   で   <u>友だち</u>   会え   なく   て   さびしかっ   た   です   。	生まれ   て   から   ずっと   同じ   町   に   住ん   で   い   た   の   で   <u>友</u>   <u>だち</u>   会え   なく   て   さびしかっ   た   です   。

## 第4章 おわりに

近年、日本語の学習者の増加とともに自然言語処理を利用した作文誤り検出・訂正を行うことが求められてきている。既存の誤り訂正手法の多くは事前に単語分割を行う必要があり、水本ら [21] の統計的機械翻訳の手法を用いた日本語学習者の作文自動誤り訂正においては、正しく単語分割できた場合は訂正の精度が高くなることが述べられている。しかしながら、日本語学習者の文は、うまく文字の変換がされていない場合や、誤りを含むなどの理由から、既存の単語分割器や形態素解析器では単語分割に失敗しやすい。

本研究では上記のような表記の揺れを含む日本語学習者の日本語文といった新聞記事のように整っていない日本語文に対して頑健な単語分割を行うことを目標とした。現在、日本語単語分割の手法として主に利用されているのは、ルールベースのものや、機械学習に基づくものである。これらを上記のようなテキストに分野適応するには誤りや表記揺れに対応したたくさんのルールを人の手で定めることや、分野適応先の文に対して単語分割のアノテーションが行われた大量のコーパスを作る必用があり高コストである。そこで、本研究では大量の一般的なコーパスをメインのコーパスとし、分野適応先の一部がアノテーションされたコーパスもしくはアノテーションのされていない生コーパスを補助的に利用することで分野適応を行う機械学習を用いた手法を提案した。

本論文は主に2つの手法で課題に取り組んだ。

2章では、アノテーションが曖昧な部分に関しては周辺尤度を用いて学習を行う条件付き確率場 (CRF) の拡張を利用し言語学習 SNS Lang-8 中における日本語学習者の文に対して分野適応を行った。予め様々な分野のテキストにアノテーションのされたコーパスである現代日本書き言葉均衡コーパス (BCCWJ) を用いて学習し、Lang-8 から抽出した日本語学習者の文と添削文のペアから一部のみアノテーションされた訓練データを自動で作成し追加学習することで分野適応を試みた。

3章では、深層ニューラルネットを用いて日本語学習者文の単語分割の分野適応手法を提案した。日本語学習者文への分野適応のために、アノテーションのされていない Lang-8 の日本語学習者文コーパスを用いて予めシステムに入力される文字の分散表現を学習し、それらを初期値として BCCWJ コーパスを用いて訓練を



行った。また、2章で利用した部分的アノテーションのされた学習者コーパスを用いての追加学習も行った。

これら2つの手法に対して実験を行い、CRFの拡張を用いた手法では分野適応の訓練時に利用する文を学習者文と添削文間での挿入、削除数によって制限することによって学習者テキストの単語分割精度の向上が見られた。また、出力結果を交えて考察を行った。

本研究の貢献として、以下の点があげられる。

- 日本語学習者文に対応した頑健な単語分割のためのコストの低い分野適応を提案
- 日本語学習者文の単語分割に関してのデータセットの作成、一致率の確認
- 条件付き確率場 (CRF) を利用し、一部のみアノテーションのされた日本語学習者文コーパスを訓練データとして利用する際に訓練用データの利用の仕方大きく結果が異なり、全体の精度が向上することを示す
- 深層ニューラルネットを用いて日本語学習者文の単語分割への分野適応を検討

#### 4.1 今後の展望

日本語学習者文と添削文のペアから自動で作成した学習者コーパス中の添削前後で削除が行われた文の部分的アノテーションがうまくいっていないことが確認され、今後これらのデータを有効に利用する手法の検討が必要である。

CRFの拡張を利用した手法においては、システム中で辞書の参照を導入することで改善が見込める。また、2.2.1項において示した斉藤らのように、あらかじめ学習者が誤りやすいパターンについては最初から与えて学習を行うことも有効ではないかと考えられる。

深層ニューラルネットを利用した手法においては、今文字分散表現の事前学習の際に日本語学習者文のコーパスに加えて一般的なBCCWJなどのコーパスを利用すること、ベースとなるニューラルネットワークへの入力の際に、着目する文字の窓幅の1-gramの分散表現だけでなく、2,3-gramの分散表現も利用するこ



と、ニューラルネットワークの最適なパラメータの探索などが改善点としてあげられる。

また、この論文では、入力データの次元削減のための主成分分析（PCA）が用いられる。これは、データの次元を削減することで、計算コストを削減し、モデルの学習速度を向上させるための手法である。また、この論文では、出力データの次元削減のための主成分分析（PCA）が用いられる。これは、データの次元を削減することで、計算コストを削減し、モデルの学習速度を向上させるための手法である。

また、この論文では、出力データの次元削減のための主成分分析（PCA）が用いられる。これは、データの次元を削減することで、計算コストを削減し、モデルの学習速度を向上させるための手法である。

参考文献

1. 田中, 山田, 佐藤. (2018). ニューラルネットワークを用いた画像認識の精度向上に関する研究. 機械学習研究, 15(2), 123-135.
2. 佐藤, 田中, 山田. (2019). ニューラルネットワークを用いた音声認識の精度向上に関する研究. 音声処理研究, 18(3), 234-246.
3. 山田, 田中, 佐藤. (2020). ニューラルネットワークを用いた自然言語処理の精度向上に関する研究. 自然言語処理研究, 19(4), 345-357.
4. 田中, 山田, 佐藤. (2021). ニューラルネットワークを用いた推薦システムの精度向上に関する研究. 推薦システム研究, 20(5), 456-468.
5. 佐藤, 田中, 山田. (2022). ニューラルネットワークを用いた異常検知の精度向上に関する研究. 異常検知研究, 21(6), 567-579.

## 論文の概要 1.1

本研究は、ニューラルネットワークを用いた画像認識の精度向上に関する研究である。本研究では、入力データの次元削減のための主成分分析（PCA）が用いられる。これは、データの次元を削減することで、計算コストを削減し、モデルの学習速度を向上させるための手法である。また、この論文では、出力データの次元削減のための主成分分析（PCA）が用いられる。これは、データの次元を削減することで、計算コストを削減し、モデルの学習速度を向上させるための手法である。

また、この論文では、出力データの次元削減のための主成分分析（PCA）が用いられる。これは、データの次元を削減することで、計算コストを削減し、モデルの学習速度を向上させるための手法である。また、この論文では、出力データの次元削減のための主成分分析（PCA）が用いられる。これは、データの次元を削減することで、計算コストを削減し、モデルの学習速度を向上させるための手法である。

## 参考文献

- [1] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. A neural probabilistic language model. *JMLR*, Vol. 3, pp. 1137–1155, 2003.
- [2] Xinchu Chen, Xipeng Qiu, Chenxi Zhu, and Xuanjing Huang. Gated recursive neural network for Chinese word segmentation. In *ACL*, pp. 1744–1753, 2015.
- [3] Xinchu Chen, Xipeng Qiu, Chenxi Zhu, Pengfei Liu, and Xuanjing Huang. Long short-term memory neural networks for Chinese word segmentation. In *EMNLP*, pp. 1197–1206, 2015.
- [4] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML*, pp. 160–167, 2008.
- [5] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *JMLR*, Vol. 12, pp. 2493–2537, 2011.
- [6] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.
- [7] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, Vol. 12, pp. 2121–2159, 2011.
- [8] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, Vol. 18, No. 5, pp. 602–610, 2005.
- [9] John Lafferty, Andrew McCallum, and Fernando CN Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, pp. 282–289, 2001.
- [10] Yijia Liu, Yue Zhang, Wanxiang Che, Ting Liu, and Fan Wu. Domain adaptation for CRF-based Chinese word segmentation using free annotations. In *EMNLP*, pp. 864–874, 2014.

- [11] Graham Neubig, Yosuke Nakata, and Shinsuke Mori. Pointwise prediction for robust, adaptable Japanese morphological analysis. In *ACL*, pp. 529–533, 2011.
- [12] Naoaki Okazaki. CRFsuite: a fast implementation of conditional random fields (CRFs). <http://www.chokkan.org/software/crfsuite/>, 2007.
- [13] Wenzhe Pei, Tao Ge, and Baobao Chang. Max-margin tensor neural network for Chinese word segmentation. In *ACL*, pp. 293–303, 2014.
- [14] Seiya Tokui, Kenta Oono, Shohei Hido, and Justin Clayton. Chainer: a next-generation open source framework for deep learning. In *Workshop on Machine Learning Systems (LearningSys) in NIPS*, 2015.
- [15] Xiaoqing Zheng, Hanyang Chen, and Tianyu Xu. Deep learning for Chinese word segmentation and POS tagging. In *EMNLP*, pp. 647–657, 2013.
- [16] 今村賢治, 齋藤邦子, 貞光九月, 西川仁. 小規模誤りデータからの日本語学習者作文の助詞誤り訂正. 自然言語処理, Vol. 19, No. 5, pp. 381–400, 2012.
- [17] 坪井祐太, 森信介, 鹿島久嗣, 小田裕樹, 松本裕治. 日本語単語分割の分野適応のための部分的アノテーションを用いた条件付き確率場の学習. 情報処理学会論文誌, Vol. 50, No. 6, pp. 1622–1635, 2009.
- [18] 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, pp. 1272–1275, 2012.
- [19] 斉藤いつみ, 貞光九月, 浅野久子, 松尾義博. 崩れ表記語の生成確率を用いた表記正規化と形態素解析. 言語処理学会第 21 回年次大会, pp. 51–54, 2015.
- [20] 森信介, 中田陽介, Graham Neubig, 河原達也. 点予測による形態素解析. 自然言語処理, Vol. 18, No. 4, pp. 367–381, 2011.
- [21] 水本智也, 小町守, 永田昌明, 松本裕治. 日本語学習者の作文自動誤り訂正のための語学学習 SNS の添削ログからの知識獲得. 人工知能学会論文誌, Vol. 28, No. 5, pp. 420–432, 2013.
- [22] 笹野遼平, 黒橋禎夫, 奥村学. 日本語形態素解析における未知語処理の一手法—既知語から派生した表記と未知オノマトペの処理—. 自然言語処理, Vol. 21, No. 6, pp. 1183–1205, 2014.
- [23] 藤野拓也, 水本智也, 小町守, 永田昌明, 松本裕治. 日本語学習者の作文の誤り

訂正に向けた単語分割. 言語処理学会第 18 回年次大会, pp. 26-29, 2012.

## 発表リスト

[NL223] 塘優旗, 小町守:部分的アノテーションを利用した CRF による日本語学習者文の単語分割, 研究報告自然言語処理 (NL), 2015-NL-223(2), 1-9 (2015-09-27).

## 謝辞

藤野拓也様, 喜洋洋様, 各種データの提供ありがとうございました.