

学修番号 15890515

修士論文

LSTM を用いた日本語形態素解析

北川 善彬

2017 年 2 月 24 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

北川 善彬

審査委員：

小町 守 准教授 (主指導教員)

高間 康史 教授 (副指導教員)

片山 薫 准教授 (副指導教員)

LSTM を用いた日本語形態素解析*

北川 善彬

内容梗概

日本語の処理において形態素解析は機械翻訳、対話などの後段の処理のために必要となる基本的なタスクである。日本語の形態素解析では、主な処理として、単語分割と品詞タグ付けが行われるのが一般的である。日本語、中国語のようなスペースなどの区切り文字のない言語においては、形態素解析のエラーによる後段のタスクへの影響は無視できない。形態素解析は教師データを用いた系列ラベリングによる手法や条件付き確率場を用いた手法が主流であるが、素性を人手で作成する必要がありコストがかかり、素性がスパースになりやすい傾向がある。

最近の研究では、自然言語処理のタスクに対して、ニューラルネットワークのモデルの適用が盛んに研究されている。ニューラルネットワークのモデルは、隠れ層の数やベクトルの次元といったハイパーパラメータのチューニングの問題を伴うが、以前のような素性エンジニアリングによる手間を軽減し、高次元でスパースな素性ではなく、低次元で密な素性による学習を実現している。中国語の単語分割においては、ニューラルネットワークを利用した単語分割が *state-of-the-art* を記録した。この要因として、Long Short-Term Memory (LSTM) により系列全体の情報や複数の文字から作られる文字の N-gram などのスパースな素性をうまく扱えるようになった点が考えられる。

日本語の形態素解析では、リカレントニューラルネットワーク言語モデル (RNNLM) を利用した研究があるものの、ニューラルネットワークの構造を用いた形態素解析の研究はなされていない。このような背景から、本論文では深層ニューラルネットワークを利用した日本語形態素解析に関する分析を行った。本研究では、深層ニューラルネットワークによる手法を日本語に適用するために、ひらがな、カタカナ、漢字といった日本語特有の入力情報と日本語形態素解析で広く

*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 15890515, 2017 年 2 月 24 日.

用いられる辞書を組み込む手法を提案した。

本研究では、先行研究に従い、初めに単語分割を行い、その後に分割された単語の品詞を推定するというカスケード方式で形態素解析を実現した。これらの2つのステップでは、どちらも系列ラベリングによる手法を採用した。つまり、単語分割においては、それぞれの文字に、B (Begin), I (Inside), E (End), S (Single) のいずれかのラベルを付与するタスクを解き、品詞付与においては、単語分割後のそれぞれの単語に、名詞、動詞、形容詞などの品詞を付与するタスクを解くことで形態素解析を実現した。

実験において、データとしては、日本語解析で広く用いられている日本語書き言葉均衡コーパス (BCCWJ) と京大コーパスを使用した。BCCWJ は日本語の様々なジャンルのテキストに、京大コーパスは毎日新聞に、単語境界と品詞情報等がそれぞれアノテーションされたコーパスである。単語分割に関しては、単語の適合率と再現率による F 値、品詞付与に関しては、単語/品詞のペアに対しての適合率と再現率による F 値で評価した。すなわち、品詞付与の評価では、単語分割があつていてかつ品詞付与が正しくなければ正解にならない。また、先行研究の手法と本手法の違い、入力情報の比較、BCCWJ のジャンルによる比較、実際の出力から考察を行った。

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、貢献、概要を述べる。第 2 章では深層ニューラルネットワークを利用した単語分割、品詞タグ付け (POS 付与) についての関連研究について述べる。第 3 章では深層ニューラルネットワークを利用した日本語形態素解析を単語分割、品詞付与に分けて解く手法について述べる。第 4 章では単語分割と品詞付与の実験結果と考察を行う。第 5 章では本研究の結論と今後の展望について述べる。

Long Short-Term Memory for Japanese Morphological Analysis*

Yoshiaki Kitagawa

Abstract

This paper presents a Long Short-Term Memory (LSTM) neural network approach to Japanese Morphological Analysis (JMA). Previous work in Chinese word segmentation (CWS) has succeeded in using recurrent neural networks such as LSTM and gated recurrent unit (GRU) and achieves state-of-the-art accuracy. Unlike Chinese, Japanese has several character types such as hiragana, katakana, and kanji, that produce orthographic variations and make word segmentation even difficult. Also, it is important for JMA task to consider the whole sequence to correctly segment, yet traditional JMA approaches rely on features in a fixed window. To address this problem, we propose to employ LSTM to JMA. Experimental results show that our proposed model outperformed state-of-the-art method only in Japanese Word Segmentation.

*Master's Thesis, Department of Information and Communication Systems, Graduate School of System Design, Tokyo Metropolitan University, Student ID 15890515, February 24, 2017.

目次

図目次	vi
第 1 章 はじめに	1
第 2 章 関連研究	3
2.1 単語分割の関連研究	3
2.2 品詞タグ付けの関連研究	4
第 3 章 日本語形態素解析におけるニューラルモデル	5
3.1 ニューラルネットの構造	5
3.2 LSTM (Long Short-Term Memory)	8
3.3 ニューラルネットの入力 (素性)	8
3.3.1 文字レベルの入力	8
文字ベクトルによるベースライン	9
文字種ベクトル (提案手法)	9
文字 N-gram ベクトル (提案手法)	10
3.3.2 単語レベルの入力	10
単語の辞書ベクトル (提案手法)	10
単語/品詞の辞書ベクトル (提案手法)	12
3.4 学習	12
第 4 章 実験と考察	14
4.1 実験	14
4.1.1 評価	14

4.1.2	データ	14
4.1.3	実験設定	15
4.1.4	ツール	15
4.1.5	ハイパーパラメータ	16
4.1.6	結果	16
	単語分割	16
	品詞付与	17
4.2	考察	17
4.2.1	コーパスによる比較	18
4.2.2	手法の比較	19
4.2.3	入力情報の比較	20
4.2.4	解析結果の比較	21
第 5 章	おわりに	22
	謝辞	23
	参考文献	24
	発表リスト	27

図目次

3.1	単語分割における系列ラベリングの図	6
3.2	品詞付与における系列ラベリングの図	7
3.3	ニューラルネットワーク全体の構造	7
3.4	辞書ベクトルの例	11

第 1 章 はじめに

日本語の形態素解析は、日本語処理の基本タスクの一つである。日本語においては、単語分割とその単語の追加情報として品詞などの付加情報を含めて解析する形態素解析が主流である。特に、日本語や中国語などの単語分割の誤りは、自然言語処理の機械翻訳や対話などの下流アプリケーションに問題を引き起こす可能性がある。したがって、正確な形態素解析を行うことは非常に重要である。

高精度な解析を実現するために、日本語形態素解析 (JMA) における手法のほとんどすべてが、洗練された素性エンジニアリングを有する識別モデルを利用する。しかし、機械学習ベースの手法は、人手作業による素性テンプレートを必要とし、データスパースネスの問題に悩まされる傾向がある。素性エンジニアリングの問題に対処するために、ニューラルネットワークモデルはさまざまな NLP タスクで研究されてきた [1, 2, 3, 4, 5]。ニューラルネットワークモデルは、表現学習によって学習された *embeddings* と呼ばれる密な特徴ベクトルの使用を可能にすることができる [6]。

日本語形態素解析のもう一つの重要な問題は、文脈のモデル化である。従来の日本語形態素解析の手法では、固定ウィンドウ内のローカルな素性を展開した素性テンプレートを使用している。しかし、このような手法ではウィンドウの外側にあるグローバルな情報は除外されることになる。それに対し、リカレントニューラルネットワークモデル (RNN) は Long Short-Term Memory (LSTM) を利用することで長距離情報をとられることを可能にし、中国語の単語分割において *state-of-the-art* の精度を記録している [7]。しかし、LSTM のアプローチが日本語形態素解析においても有効であるかどうかは自明でない。なぜなら、日本語では、ひらがな、漢字、カタカナといった様々な文字種が存在するからである。

このような背景から、文字レベルの埋め込みと長距離依存を組み込んだ日本語形態素解析のための LSTM ネットワークモデルを提案する。この研究の主な貢献は以下である。

- LSTM を用いた日本語形態素解析を提案し、文字 N-gram などの疎な特徴をどのように利用するかを検討した。
- 実験において、単語分割に対しては、提案手法がいくつかのデータセットで

単語レベルと文レベルの両方の精度で最高精度を達成することを示した.

第 2 章 関連研究

2.1 単語分割の関連研究

日本語の形態素解析 (JMA) とは、一般に、単語分割と品詞タグ付けを行うことを指す。日本語形態素解析においては、教師あり学習による手法が広く使われている。一つがよく知られた手法として、辞書を使ってラティスを作り、条件付き確率場 (CRF) でコストを計算することにより単語分割と品詞タグ付けを同時に解くことで形態素解析を行う手法 [8, 9] がある。この手法は、単語の系列を考慮して正確な結果を得ることは知られているが、トレーニングデータの性質がテストデータと異なる場合に頑健でないという問題がある。もう一つがよく知られた手法として、周辺の単語や文字の情報から SVM 等の線形分類器により、単語分割をし、その後、品詞等の情報を推定することで形態素解析を行うカスケード方式による手法 [10, 11] が存在する。この手法は、違ったドメイン間で頑健に動くことが知られており、部分的にアノテーションされたコーパスを利用して学習することもできる。しかし、両方のアプローチでは、ローカルな固定長の入力を使用するため、グローバルな文脈を考慮できない問題がある。また、どちらの手法も入力情報のスパースネスの問題に悩まされている。

一方、中国語の単語分割において、深層ニューラルネットワークを用いた手法が盛んに研究されている。[7, 12, 13]。しかし、深いニューラルネットワークのアプローチは、以前のアプローチと比較して高い計算コストを必要とする傾向がある。日本語形態素解析においても、既存のアプローチに再帰ニューラルネットによる言語モデル (RNNLM) を統合した形態素解析手法が提案されている [14]。それに対し、本研究では、既存の手法に追加情報として リカレントネットワーク (RNN) を使うのではなく、長短期記憶ユニット (LSTM) を用いて日本語の形態素解析を直接学習する。

2.2 品詞タグ付けの関連研究

品詞タグ付けは日本語の形態素解析においては、単語分割と同時に解かれることが多い [8, 9]. 一方で、中国語においては単語分割を単体で解くことが多い [7, 12, 13, 15]. このように、中国語では単語分割をニューラルネットワークを用いて解く研究は盛んであるが、品詞タグ付けの研究が盛んに行われていない現状がある. Neubig ら [10, 11] の日本語形態素解析の研究にあるように、単語分割と品詞タグ付けに必要な情報には共通部分が多い. そこで、本研究では、Neubig らの研究をもとに単語分割と品詞分割の両方をニューラルネットワークで解く手法を提案する.

中国語、日本語と異なり、単語がスペースで区切られている英語などの言語においては、品詞タグ付けをニューラルネットを利用して解く研究が古くから行われている [16, 17, 18]. Schmid [17] はニューラルネットによる品詞タグ付けが通常の隠れマルコフモデル (HMM) よりも優れた精度を上げることがを報告している. Tsuboi [18] はコーパスワイドの情報ニューラルネットに導入する手法を提案し、英語の品詞タグ付けタスクにおいて state-of-the-art の精度を記録している. そこで、本研究では、これらの研究の知見を文字種や辞書などの情報を多く利用する日本語形態素解析に適用する手法を提案する.

第 3 章 日本語形態素解析におけるニューラルモデル

3.1 ニューラルネットの構造

本研究では、単語分割を行った後、品詞タグ付けを行うカスケード方式で日本語形態素解析を実現する。これらの二つのタスクは、どちらも系列ラベリング問題としてみなすことが可能である。また、両者の間では、入力として必要とする情報に共通部分が多い。表 3.1 は単語分割と品詞付与における素性をまとめたものである。

単語分割に関して、それぞれの文字は $\{B, I\}$, $\{B, I, S\}$, $\{B, I, E, S\}$ などのラベルがつけられる。ここで、B は Begin, I は Inside, E は End, S は Single を表す。本研究では上に挙げた 3 通りのラベル付けを採用する。入力と出力の関係を図 3.1 に示す。

品詞付与に関して、それぞれの単語は、名詞、動詞などの品詞がラベルづけされる。入力と出力の関係を図 3.2 に示す。

また、本節では、日本語形態素解析におけるニューラルモデルの基本的な構造に関しての説明を行う。ネットワークの全体像を図 3.3 に示す。ここで、図 3.3 は単語分割における構造を示しており、品詞付与については、本文で説明する。

初めに、フィードフォワードニューラルネットワーク（以下、**FFNN**）について説明する。文字数が n の文 $c_{1:n}$ が与えられたとき、ウィンドウサイズを k (k : 奇数) とすると、ある文字 c_t ($1 \leq t \leq n$) に対する入力は、 $(c_{t-\frac{k-1}{2}}, \dots, c_t, \dots, c_{t+\frac{k-1}{2}})$ となる。また、文頭と文末に文頭記号、文末記号を付加する。また、品詞付与に関して、単語数が n の文 $w_{1:n}$ が与えられたとき、ウィンドウサイズを k (k : 奇数) とすると、ある単語 w_t ($1 \leq t \leq n$) に対する入力は、 $(l_{\frac{k-1}{2}}, \dots, l_1, w_t, r_1, \dots, r_{\frac{k-1}{2}})$ となる。ここで、 l_s は単語の左側の s 番目の文字を表し、 r_s は単語の右側の s 番目の文字を表す。

ニューラルネットワークを用いるための最初のステップとして、文字を実数値のベクトルにする必要がある。これを *embedding* と呼ぶ [19]。各文字の *embedding* は *lookup table* から取り出され、それらを連結することで入力ベクトル $\mathbf{x} \in \mathbb{R}^{H_1}$ を得る。ここで、 H_1 は入力文字列の *embedding* 層のサイズであり、その値は、 $k \times d$ である。また、 d は各文字の *embedding* の次元数である。

表 3.1 単語分割と品詞付与における素性

タスク	タイプ	素性
単語分割と品詞付与	文字レベル	文字の N-gram 文字種の N-gram
単語分割のみ	単語レベル	単語の辞書素性
品詞付与のみ	単語レベル	単語/品詞の辞書素性 単語 Uni-gram 単語の文字種



図 3.1 単語分割における系列ラベリングの図

次に入力ベクトル \mathbf{x}_t は以下の線形変換に渡され、成分ごとに *sigmoid*, *tanh* などの活性化関数 g にかけられ、 \mathbf{h}_t を得る。

$$\mathbf{h}_t = g(\mathbf{W}_1 \mathbf{x}_t + \mathbf{b}_1) \quad (3.11)$$

ここで、 $\mathbf{W}_1 \in \mathbb{R}^{H_2 \times H_1}$, $\mathbf{b}_1 \in \mathbb{R}^{H_2}$, $\mathbf{h}_t \in \mathbb{R}^{H_2}$ であり、 H_2 は (b) の隠れ層の次元である。

さらに、 $\{\text{B, I, E, S}\}$ などの出力ラベル集合 T に対して、隠れ層のベクトル \mathbf{h}_t は以下の線形変換に渡され、出力 \mathbf{y}_t を得る。

$$\mathbf{y}_t = \text{softmax}(\mathbf{W}_2 \mathbf{h}_t + \mathbf{b}_2) \quad (3.12)$$



図 3.2 品詞付与における系列ラベリングの図

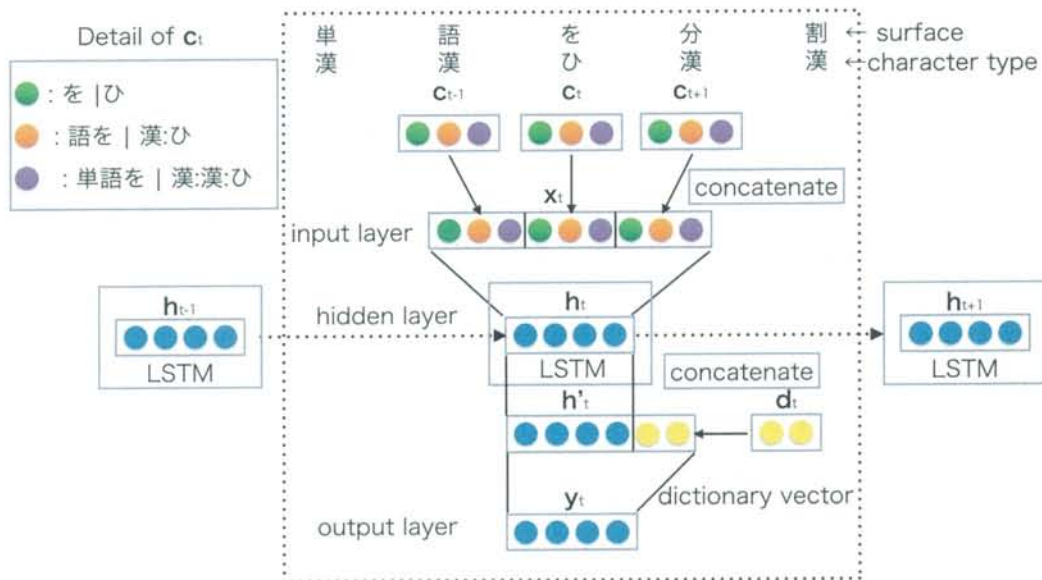


図 3.3 ニューラルネットワーク全体の構造

ここで、 $|T|$ を出力ラベルの次元として、 $\mathbf{W}_2 \in \mathbb{R}^{|T| \times H_2}$, $\mathbf{b}_1 \in \mathbb{R}^{|T|}$, $\mathbf{h}_t \in \mathbb{R}^{|T|}$ である。

次に、リカレントニューラルネットワーク（以下、**RNN**）について説明する。RNN は現在の隠れ層の入力に 1 つ前の隠れ層の値を入れることで出力層を得る、定式化すると、(3.11) 式に 1 つ前の入力を足して以下のようなになる。

$$h_t = g(Uh_{t-1} + W_1x_t + b_1) \quad (3.13)$$

ここで、 $U \in \mathbb{R}^{H_2 \times H_2}$ である。RNN は自然言語処理の様々なタスクで成功を収めているが、誤差伝搬時に系列の初めに行くに連れて勾配が伝わらなくなるという問題を抱えている。

3.2 LSTM (Long Short-Term Memory)

本節では、Long Short-Term Memory (以下、**LSTM**) について説明する [20]。LSTM は上で述べた RNN の問題を解決するための拡張である。LSTM の中心となるのは、各ステップにおける入力に対してのメモリセル c である。このセル c は入力ゲート、忘却ゲート、出力ゲートの 3 つのゲートにより制御されている。以下に LSTM の定式化を示す。ここで、 σ , ϕ はそれぞれ *sigmoid*, *tanh* であり、 \odot は成分ごとの掛け算を表すアダマール積である。

$$i_t = \sigma(W_{ix}x_t + W_{ih}h_{t-1}) \quad (3.21)$$

$$f_t = \sigma(W_{fx}x_t + W_{fh}h_{t-1}) \quad (3.22)$$

$$c_t = f_t \odot c_t + i_t \odot \phi(W_{cx}x_t + W_{ch}h_{t-1}) \quad (3.23)$$

$$o_t = \sigma(W_{ox}x_t + W_{oh}h_{t-1}) \quad (3.24)$$

$$h_t = o_t \odot \phi(c_t) \quad (3.25)$$

ここで、 σ と ϕ はそれぞれシグモイド関数と *tanh* 関数である。すべてのベクトルは、隠れ層と同じサイズである。異なる添え字を持つパラメータ行列 W はすべて正方行列で、 W_{ic} , W_{fc} , W_{oc} は対角行列である。 \odot はベクトルの要素ごとの積を表す。

3.3 ニューラルネットの入力 (素性)

3.3.1 文字レベルの入力

この節では、日本語形態素解析の文字ベクトル c_t について説明する。正式には、文字ベクトル c_t を次のように定義する。

$$\mathbf{c}_t = \mathbf{l}_t \oplus \mathbf{e}_t \quad (3.31)$$

ここで \oplus はベクトルの連結を表し、 \mathbf{l}_t と \mathbf{e}_t はそれぞれ単なる文字 embeddings と文字種 embeddings を表す。これらの embeddings は、入力層に渡される。ここでは、日本語形態素解析で使用される素性について説明し、それらニューラルネットの構造にどのように組み込むかを説明する。

文字ベクトルによるベースライン

ニューラルネットワークを使って文字データを処理する最初のステップは、それらを embeddings と呼ばれる連続値のベクトルとして表現することである [21, 19].

形態素解析では、サイズ $|C|$ の文字の辞書 C を作成する。特に言及がない場合、文字の辞書はトレーニングセットから抽出され、未知の文字はコーパスで使用されていない特殊記号にマッピングされる。

素性テンプレートを使用する伝統的な機械学習アプローチでは、各文字を個別に one-hot ベクトルとして扱う。しかし、ニューラルネットワークモデルでは、embeddings と呼ばれる連続値ベクトルとして表現するのが一般的である [21, 19]. このような表現学習は、データスパースネスの問題を克服することができるため、NLP で活発に研究されているトピックの一つである。したがって、各文字を実数ベクトル $\mathbf{v}_c \in \mathbb{R}^d$ として表現するためにこの手法を用いる。 d はベクトル空間の次元数である。各文字について、 \mathbf{v}_c を埋め込んだ対応する文字は lookup tabel と呼ばれるものから取り出される。この lookup table は初めはランダムな連続値のベクトルを並べた行列であるが、学習をする際に、適当な値に更新される。

文字種ベクトル（提案手法）

文字 embeddings は接頭辞と接尾辞の識別に非常に有効である。しかし、文字 embeddings は素性がスパースになりやすい問題がある。この問題に対処するために、日本語の単語分割には、ひらがな、カタカナ、漢字などの文字種を活用することが有効であることが知られている [10]. たとえば、カタカナは借用語になりがちで、文字種から別の文字種への移行は単語境界になりやすいのである [22].

したがって、本研究では、embeddings として文字種情報を組み込む。まず、各

文字を文字タイプに対応する one-hot ベクトルに変換する。one-hot ベクトルは、ひらがな、カタカナ、漢字、アルファベット、数字、記号、開始記号、終端記号のいずれかに対応する次元に 1 が立ち、その他の次元が 0 のベクトルとして構成される。

one-hot ベクトルは、文字の embeddings と同様に文字種の embeddings に変換される。embeddings はまた、ニューラルネットワークの入力ベクトルに連結される。

文字 N-gram ベクトル (提案手法)

ニューラルネットワークは、疎なベクトルを密なベクトルに変換して扱うことができるという利点をもつ。これは、文字 tri-gram のような疎な素性を利用することを可能にする。したがって、embeddings として文字と文字の両方に対して N-gram を使用する。正確には、uni-gram だけでなく、bi-gram と tri-gram に対しても one-hot ベクトルを作成する。各 embeddings は、文字 embeddings と同様に lookup table によって取り出される。

以下のように、embeddings ベクトル l_t と e_t を定義する。

$$l_t = l_{[t-2:t]} \oplus l_{[t-1:t]} \oplus l_{[t]} \quad (3.32)$$

$$e_t = e_{[t-2:t]} \oplus e_{[t-1:t]} \oplus e_{[t]} \quad (3.33)$$

ここで、 $l_{[a:b]}$ は a から b までの文字列の embeddings を表す。 e_t についても同様である。

3.3.2 単語レベルの入力

単語の辞書ベクトル (提案手法)

この節では、単語の辞書ベクトルについて説明する。この素性は、表 3.1 にあるように単語分割にのみ適用される素性である。文字の embeddings, 文字種の embeddings, およびそれらの N-gram 拡張は、単語境界がアノテーションされたコーパスから文字ベースの素性を学習することに優れている。一方で、このような文字ベースの情報だけでは、日本語形態素解析において、どの文字列が単語を構成

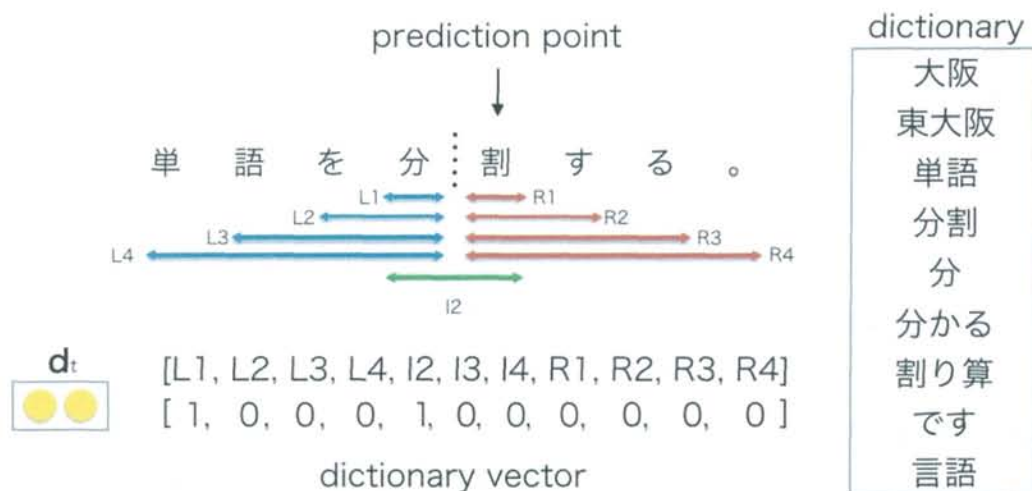


図 3.4 辞書ベクトルの例

するかを決定するのに有用な単語レベルの情報を欠いている。したがって、日本語の形態素解析は辞書を利用することが多い。辞書は、文字ベースの日本語形態素解析のアプローチでは必要ではないが、デコード時に単語ラティスを使用する日本語形態素解析にとって不可欠である。

しかし、辞書情報をニューラルネットワークの構造に組み込む方法は自明ではない。[18] は、同じレイヤー内で密なベクトル表現と疎なベクトル表現の両方を学習することは有効でないことを示唆している。したがって、辞書ベクトルの作成に関しては、embeddings にせずに [10] に従い辞書ベクトルを作成し、これを、出力層の入力として、隠れ層に連結させて使用する (表 3.3)。

図 3.4 は、辞書ベクトルの作成方法を示している。辞書ベクトルは、左側素性 L 、右側素性 R 、および内側素性 I の 3 つの部分から構成される。たとえば、予測点の左側に、長さが 2 の単語が存在する場合、 $L2$ が 1 となる。単語の長さが特定のしきい値を超えている場合は、単語の長さを特定の長さに切り捨てる。本研究では、neubig ら [10] に従い 4 をしきい値として採用している。 L と R とは異なり、 I は単語の長さが 1 を超える場合にのみ定義される。境界にまたがって長さ 2 の単語が存在する場合、 $I2$ が 1 となる。単語長が特定の閾値を超える場合は L と R と

同様であるが, $I1$ は定義できないことに注意したい. 最後に, これらを並べたものを辞書ベクトル d_t として定義する.

辞書ベクトル d_t は現在の隠れ層に連結される. 正式には, 新しい隠れ層 h'_t は次のように定義される. ここで注意したいのは, 次の隠れ層への入力として渡される LSTM 上の隠れ層は h'_t でなく h_t であることである.

$$h'_t = h_t \oplus d_t \quad (3.34)$$

単語/品詞の辞書ベクトル (提案手法)

次に, 単語/品詞の辞書ベクトルについて説明する. この素性は, 表 3.1 にあるように品詞付与にのみ適用される素性である. これは, 品詞を付与する際, 実際に, 現在注目している単語にどのような品詞がつく可能性があるかを事前に用意した辞書を利用して入力するためのものである. 辞書はトレーニングデータから存在する単語/品詞のペアを集計することで作成する. 単語/品詞の辞書ベクトルは品詞の数と同じ次元を持つベクトルである. 現在注目している単語 w_t に対し, 辞書を利用して w_t につく品詞を列挙する. 次に, それらの品詞に対応した次元が 1, それ以外が 0 になっているベクトル p_t を作成する. このベクトルは単語の辞書ベクトルと同様に現在の隠れ層に連結される.

$$h'_t = h_t \oplus p_t \quad (3.35)$$

3.4 学習

本研究では, KyTea [10] のように系列中のある文字に対して, 周辺の文字等の情報からラベルを推定する点推定による学習を検討した. 図 3.3 における出力 y_t に対し, 単語分割においては文字 c_t に対して, また, 品詞付与については単語 w_t に対して正解ラベル分布 l_t を用意し, 次式の交差エントロピー誤差によって目的関数を設定する.

$$loss = \sum_t -l_t \log y_t + \frac{1}{2} \lambda \|\theta\|_2^2 \quad (3.41)$$

ここで、 λ は L2 正則化のハイパーパラメータであり、 θ はモデルのすべてのパラメータを表す。

第4章 実験と考察

4.1 実験

本研究では、いくつかの日本語コーパスで LSTM を利用した日本語形態素解析と Neubig ら [10] による手法を評価した。ニューラルネットワークの構造を評価するために、フィードフォワードネットワーク (FFNN) とリカレントニューラルネットワーク (RNN) を準備する。FFNN は図 3.3 の点線囲まれた部分で示されている。また、RNN は LSTM と同じ入力を使用するが、LSTM ユニットは使用しない。本研究では、実験は 2 つの部分に分けた。まず、ニューラルネットワークのアーキテクチャに関する結果と入力情報の違いによる比較を従来の手法と比べることにより行った。次に、異なる分割基準でアノテーションされた新聞コーパスについて、本手法を評価した (表 4.4 を参照)。

4.1.1 評価

評価は適合率と再現率の調和平均である F 値によって評価した。単語分割の評価の際は、単語に関する F 値、品詞については、単語/品詞のペアに関する F 値で評価する。品詞分割の際は、単語分割と品詞付与が両方正しくないと正解としないことに注意したい。

4.1.2 データ

評価には、2 つの異なるデータセットを使用した。一つは、BCCWJ (日本語書き言葉均衡コーパス) [23]、もう一つは広く使われている日本のコーパスである京都大学コーパス (ver4.0) (KC) で評価した。BCCWJ は様々なドメインで構成されているが、KC は単一ドメインのみから構成される。これらのコーパスの詳細は、表 4.1 に示されている。本研究では、Project Next NLP* に従って学習データとテストデータを設定した。京大コーパスの分割においては Kudo ら [8, 24] に従った。

*[http:// url http://plata.ar.media.kyoto-u.ac.jp/mori/research/topics/PST/NextNLP.html](http://url http://plata.ar.media.kyoto-u.ac.jp/mori/research/topics/PST/NextNLP.html)

表 4.1 実験で使ったコーパスの詳細

ドメイン	train	test
Yahoo!知恵袋	5,880	496
Yahoo!ブログ	7,036	506
白書	5,471	496
雑誌	12,369	492
新聞	16,222	495
書籍	9,470	499
BCCWJ All	56,448	2,984
京大コーパス (KC) All	18,455	1,234

4.1.3 実験設定

単語レベルの素性について, Neubig ら [10] は外部の辞書を使用せず, トレーニングコーパスから作成した辞書を使用している. したがって, 本研究では, これに従って, 学習データのすべての単語を追加したが, [10] で説明されているように, 訓練データへのオーバーフィットを防ぐために頻度が1回のシングルトンを取り除いた. 辞書機能の効果を分析するために, トレーニングセットとテストセットの両方から作成されたより大きな辞書を再作成した. これを ゴールド辞書 とする.

4.1.4 ツール

本研究では, neubig ら [10] の研究を実装した広く用いられる日本語形態素解析ツール KyTea (ver.0.4.6)[†]. 本研究では, KyTea のモデルを学習用のスクリプトを用いて学習した. 事前に訓練された KyTea のモデルは, BCCWJ から拡張された独自の単語分割基準を採用しているため, 公正な比較を行うために KyTea のモデルを再学習した.

また, FFNN, RNN, LSTM を含むニューラルネットワークベースの日本語形

[†]<http://www.phontron.com/kytea/index.html>[10]

態素解析モデルを、Chainer (ver 1.4.0)[‡] [25] を使用して実装した。

4.1.5 ハイパーパラメータ

予備実験により、Chen ら [7] の研究をもとにさらに良いハイパーパラメータを探索した。本研究で使用されているパラメータは、表 4.2 に示されている。

ウィンドウサイズ. 予備実験では、ウィンドウサイズ 5 が精度または処理時間のいずれにおいても他より優れていることがわかった。ウィンドウサイズ 7 は、ウィンドウサイズ 5 から F1 スコアが大きく向上しなかったため、精度と時間とのトレードオフのために 5 を選んだ。

文字種ベクトルの次元. 本研究では、文字のベクトルの次元は Chen ら [7] に従ったが、文字種ベクトルの 6 つの次元を探索した。予備実験によれば、10 が精度または時間のいずれにおいても他のものよりも優れた性能をもたらすので、文字種ベクトルの次元を 10 にした。文字種ベクトルの次元 20 と 50 は 10 と大差はないが、時間のコストを考慮して 10 を選択した。

ラベルセット. 中国語単語分割では、ラベルセット {B, I, E, S} がよく使われる。対照的に、日本語単語分割では、様々なラベルセットが採用されている。本研究では 3 つのラベルセットを調べ、{B, I, E, S} が他のものよりわずかに優れていることをからこれを採用した。

学習率. 日本語形態素解析では、学習率は精度に大きく影響する。学習率 0.1 は他の NLP タスクの学習率よりも大きいですが、0.01 のような小さな学習率は精度を低下させ、学習時間のコストが高くなる。したがって、全体を通してすべての実験について学習率 0.1 を選択した。

4.1.6 結果

単語分割

表 4.3 と表 4.4 は、BCCWJ と京大コーパスの実験結果を示している。どちらのコーパスにおいても、LSTM に基づく方法は state-of-the-art の手法である

[‡]<http://chainer.org>

表 4.2 実験で使ったハイパーパラメータセット

ハイパーパラメータ	候補	実験で使った値
ウィンドウサイズ	1, 3, 5, 7	5
文字の次元	Chen ら [7] と同様	100
文字種の次元	1, 3, 5, 10, 20, 50	10
隠れ層の次元	Chen ら [7] と同様	150
ラベルセット	{B, I}, {B, I, S}, {B, I, E, S}	{B, I, E, S}
学習率	0.01, 0.1, 0.2	0.1
正則化係数 λ	Chen ら [7] と同様	0.0001

Neubig ら [10] を上回った。表 4.7 は、ドメイン別の 2 つの手法の結果を示している。本手法は、6 つの領域のうち、4 つの領域において、単語レベルの F1 と文レベルの精度の点でより高い精度を達成し、結果として全体的な性能が向上した。

品詞付与

表 4.5 と表 4.6 に BCCWJ における品詞付与の実験結果を示している。表 4.5 では、単語分割の正解が与えられた際にそれぞれの手法で品詞付与のみを行った場合、表 4.6 では、それぞれの手法で単語分割を行った後、品詞付与を行った場合の F1 をそれぞれ示している。実験の結果、品詞付与においては、単語分割も同時に解いた場合は、Neubig ら [10] の手法よりも低い精度となったが、単語分割の正解が与えられている条件では、Neubig ら [10] と同程度の精度となった。

4.2 考察

本節では、単語分割における結果に対し、コーパス、手法、入力情報、解析結果の観点から比較し考察を行った。

表 4.3 BCCWJ における単語分割の F1.

Methods	F1
FFNN	96.53
RNN	96.46
LSTM	97.00
LSTM + 文字種	97.25
LSTM + 文字種 + 単語の辞書	97.37
LSTM + 文字種 + N-gram	98.41
LSTM + 文字種 + N-gram + 単語の辞書	98.42
LSTM + 文字種 + N-gram + ゴールド辞書	98.67
KyTea 0.4.6	98.34

表 4.4 京大コーパスにおける単語分割の F1.

Methods	F1
LSTM + 文字種 + N-gram + 単語の辞書	96.47
KyTea 0.4.6	96.21

表 4.5 BCCWJ における品詞付与の F1. (単語分割が正解を与えられた場合)

Methods	F1
This work	96.72
KyTea 0.4.6	96.74

4.2.1 コーパスによる比較

本手法の特徴を知るために、本手法を KyTea と様々な分野で比較してエラー分析を行った。BCCWJ の各ドメインごとに F1 を計算し、単語分割の誤った文の数を数えた。ここで、間違った文とは、単語分割が一つ以上間違った場合間違いと

表 4.6 BCCWJ における品詞付与の F1. (単語分割も未知の場合)

Methods	F1
This work	95.24
KyTea 0.4.6	95.72

みなすものとする。表 4.7 は、本手法と KyTea の単語レベルと文レベルの比較をまとめたものである。誤った文数について、A+B の表記で誤った文の数が示されているが、A は両方の手法とも間違った文数、B がその手法のみ間違った文数を示す。ここでは F1 で大きなマージンを示したペアを選んで考察する。すなわち、白書と雑誌を分析する。

白書。このドメインは、政府によって公表された公式文書で構成されている。このため、漢字がコーパスの大部分を占めている。また、1 文当たりの文字数が多い傾向がある。このドメインでは、本手法は、F1 と誤った文の数の両方によって [10] よりも劣っている。

雑誌。このドメインには、形式的表現だけでなく口語表現も含まれている。口語表現のため、このコーパスはひらがなの割合が多い。このドメインの F1 は、両方の手法で最低である。このことから、ひらがなはパフォーマンスを低下させることがわかる。しかし、このドメインのマージンが最大であることから、文脈情報のモデル化がうまく考慮されていると考えられる。なぜなら、ひらがなの系列は分割に必要な情報がローカルウィンドウサイズ外になる傾向があるからである。

全体的に、本手法は、異なる文字種を含む複合語 (Fami ポート) では頑健である傾向があるが、[10] の方が単一の文字種 (ポストドクター) で構成された単語においては正しく分割できる傾向にあった。

4.2.2 手法の比較

本研究では、ニューラルネットワークの構造として、文字、文字種 embedding のベクトルを concat した入力ベクトルを用い、FFNN, RNN, LSTM の構造がどの程度貢献するかを評価した。表 4.3 を見ると、文字 embedding のみを入力とし

表 4.7 BCCWJ の 6 つのドメインにおける単語レベル・文レベルの評価.

ドメイン	KyTea 0.4.6		本研究	
	F1	誤った文数	F1	誤った文数
Yahoo! 知恵袋	98.38	50+25	98.44	50+ 19
白書	99.20	60+ 21	99.08	60+24
Yahoo! ブログ	99.75	76+22	99.73	76+ 21
書籍	98.15	63+ 19	98.28	63+28
雑誌	96.70	73+29	97.25	73+ 17
新聞	98.19	60+36	98.46	60+ 15
All	98.34	382+152	98.42	382+ 124

表 4.8 単語分割の誤り事例.

本手法のみ不正解 正解	エルマー とりゅう の 絵 で エルマー とりゅう の 絵 で
本手法のみ不正解 正解	うち がまんま その 環境 です 。 うち がまんま その 環境 です .
KyTea のみ不正解 正解	七百 六十 一 の ため 池 など 被害 七百 六十 一 の ため池 など 被害
KyTea のみ不正解 正解	思う とうんざり です . 思う とうんざり です .

た場合は, FFNN, RNN にあまり違いはないが, LSTM は 0.5 ポイント程度 F 値の向上に貢献した. このことから, 前の系列の文字情報は重要であるが, 必要な情報を取捨選択する必要があることを示している.

4.2.3 入力情報の比較

表 4.3 の結果をもとに, 入力情報である文字種, 単語の辞書素性, N-gram embeddings について比較を行う. 文字種. LSTM と LSTM + 文字種を比較すると, F1 は 0.25 ポイント向上する. この結果は, 文字種のベクトルが日本語形態素解析で有用であることを示している. しかし, 文字種ベクトルを使用する利点

は、ニューラルネットのアーキテクチャを LSTM に変更する場合よりも小さくなる。これは、アーキテクチャーの選択が最終的な精度に大きな影響を与えることを示唆している。

単語の辞書素性. LSTM + 文字種に辞書機能を追加すると、F1 は 0.37 向上する。この結果は、辞書の特徴が日本語形態素解析において有効であることを示している。しかし、辞書の素性を LSTM + 文字の種類 + N-gram に追加した結果にそれほど大きな違いはない。辞書は訓練コーパスから作られているので、文字ベースの N-gram embeddings が辞書の特徴を包含していると考えられる。テストコーパスから作成したゴールド辞書を使用した追加の実験では、この仮説を支持している。[§]
N-gram embedding. LSTM + 文字種と LSTM + 文字種 + N-gram を比較すると、N-gram embeddings は、本手法のパフォーマンスを大幅に向上させる。CRF や SVM などの従来の機械学習ベースのアプローチでは、このような疎な素性を利用することはできないが、LSTM ベースの提案モデルはこの情報をうまく活用している。

4.2.4 解析結果の比較

KyTea では正解できるが本手法では解析を誤る例、本手法では正解できるが、KyTea では解析を誤る例を表 4.8 に示す。本手法の特徴として、表 4.8 の 3 つ目の例のように、カタカナや漢字が混在する単語の分割に成功しているが、その他の例のように、漢字のみで構成された熟語の分割をミスしたり、ひらがなでできた名詞、形容詞などが機能表現と繋がってしまう例が KyTea よりも多く見られた。

しかし、このように傾向を推測することは可能であるが、推測した傾向と反する結果を示すものも見られることもあり、本研究を解析結果から定性的な分析をすることは困難であると考えられる。

[§] ゴールド辞書を作成するときにコーパス内のシングルトンを削除したので、テストコーパスにはまだゴールド辞書にない単語が含まれている可能性がある。

第 5 章 おわりに

本研究では，日本語形態素解析に対して LSTM を適用する手法を提案した．本研究は，文字の種類や文字の N-gram などの日本語固有の特徴をニューラルネットに組み込む手法を提案し，その手法が単語分割において，state-of-the-art の精度を達成していることを示した．これは，他の系列ラベリングタスクにも応用できると考えられる．

日本語形態素解析の現在の問題は，会話や Web テキストで頻繁に見られる口語表現に対処することである [26, 27, 28]．CNN のようなニューラルネットの構造を利用し，頑健な文字や単語表現を学習することが今後の展望である．

謝辞

僕の人生において、小町研究室に入ったことは一つの大きな転機でした。情報系の研究者や開発者として、身につけるべき基礎を学ぶことができました。それ以上にこれからの人生をどう生きるかの指針をもらったような気もしています。また、研究においては、レベルの高い環境で指導を受けることができ、非常に有意義な3年間でした。このような環境を与えてくれた指導教員の小町守先生をはじめ、研究室のメンバー全員に感謝します。

また、研究を始めた学部4年の頃、Project Next NLP を通してお会いした、東北大学の岡崎直観先生、NAIST の荒牧英治先生には、外部の学生にも関わらず、度々ミーティングを開いて研究の指導をしていただきました。結果、国際会議での発表経験をすることもでき、非常に感謝しています。さらに、修士1年、2年においては、指導教員の小町先生の紹介で、研究のインターン、開発のインターンを経験することができました。メンターとして指導くださった NTT 研究所の貞光九月さん、Microsoft Development の呉先超さんに感謝します。

さらに、本論文に関して、副査として自然言語処理以外の見識を持った建設的なコメントや意見をくださった、本学の高間康史先生、片山薫先生に感謝致します。

最後に、研究生生活を生活面、学費面でサポートしてくれた家族に感謝します。

参考文献

- [1] P. Liu, X. Qiu, X. Chen, S. Wu, and X. Huang, “Multi-timescale long short-term memory neural network for modelling sentences and documents,” EMNLP, pp.2326–2335, 2015.
- [2] I. Sutskever, O. Vinyals, and Q.V. Le, “Sequence to sequence learning with neural networks,” NIPS, pp.3104–3112, 2014.
- [3] R. Socher, J. Bauer, C.D. Manning, and A.Y. Ng, “Parsing with compositional vector grammars.,” ACL, pp.455–465, 2013.
- [4] J. Turian, L. Ratinov, and Y. Bengio, “Word representations: a simple and general method for semi-supervised learning,” ACL, pp.384–394, 2010.
- [5] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” NIPS, pp.3111–3119, 2013.
- [6] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” NIPS, pp.3111–3119, 2013.
- [7] X. Chen, X. Qiu, C. Zhu, P. Liu, and X. Huang, “Long short-term memory neural networks for Chinese word segmentation,” EMNLP, pp.1197–1206, 2015.
- [8] T. Kudo, K. Yamamoto, and Y. Matsumoto, “Applying conditional random fields to Japanese morphological analysis,” EMNLP, pp.230–237, 2004.
- [9] M. Sassano, “An empirical study of active learning with support vector machines for japanese word segmentation,” ACL, pp.505–512, 2002.
- [10] G. Neubig, Y. Nakata, and S. Mori, “Pointwise prediction for robust, adaptable Japanese morphological analysis,” ACL-HLT, pp.529–533, 2011.
- [11] G. Neubig and S. Mori, “Word-based partial annotation for efficient corpus construction,” LREC, pp.2723–2727, 2010.
- [12] X. Chen, X. Qiu, C. Zhu, and X. Huang, “Gated recursive neural network for Chinese word segmentation,” ACL-IJCNLP, pp.1744–1753, 2015.
- [13] W. Pei, T. Ge, and B. Chang, “Max-margin tensor neural network for

- Chinese word segmentation,” ACL, pp.293–303, 2014.
- [14] H. Morita, D. Kawahara, and S. Kurohashi, “Morphological analysis for unsegmented languages using recurrent neural network language model,” EMNLP, pp.2292–2297, 2015.
 - [15] J. Ma and E. Hinrichs, “Accurate linear-time Chinese word segmentation via embedding matching,” ACL-IJCNLP, pp.1733–1743, 2015.
 - [16] N. Masami, M. Katsuteru, K. Takeshi, and S. Kiyohiro, “Neural network approach to word category prediction for english texts.,” COLING, pp.213–218, 1990.
 - [17] H. Schmid, “Part-of-speech tagging with neural networks.,” COLING, pp.172–176, 1994.
 - [18] Y. Tsuboi, “Neural networks leverage corpus-wide information for part-of-speech tagging,” EMNLP, pp.938–950, 2014.
 - [19] R. Collobert and J. Weston, “A unified architecture for natural language processing: Deep neural networks with multitask learning,” ICML, pp.160–167, 2008.
 - [20] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, vol.9, no.8, pp.1735–1780, 1997.
 - [21] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, “A neural probabilistic language model,” J. Mach. Learn. Res., vol.3, pp.1137–1155, 2003.
 - [22] M. Nagata, “A part of speech estimation method for japanese unknown words using a statistical model of morphology and context,” ACL, pp.277–284, 1999.
 - [23] K. Maekawa, M. Yamazaki, T. Ogiso, T. Maruyama, H. Ogura, W. Kashino, H. Koiso, M. Yamaguchi, M. Tanaka, and Y. Den, “Balanced corpus of contemporary written japanese,” Language Resources and Evaluation, vol.48, no.2, pp.345–371, 2014.
 - [24] K. Uchimoto, S. Sekine, and H. Isahara, “The unknown word problem: a morphological analysis of japanese using maximum entropy aided by a dictionary,” EMNLP, pp.91–99, 2001.

- [25] S. Tokui, K. Oono, S. Hido, and J. Clayton, “Chainer: a Next-Generation open source framework for deep learning,” NIPS Workshop, 2015.
- [26] I. Saito, K. Sadamitsu, H. Asano, and Y. Matsuo, “Morphological analysis for Japanese noisy text based on character-level and word-level normalization,” COLING, pp.1773–1782, 2014.
- [27] R. Sasano, S. Kurohashi, and M. Okumura, “A simple approach to unknown word processing in Japanese morphological analysis,” IJCNLP, pp.162–170, 2013.
- [28] N. Kaji and M. Kitsuregawa, “Accurate word segmentation and pos tagging for Japanese microblogs: Corpus annotation and joint modeling with lexical normalization,” EMNLP, pp.99–109, 2014.

発表リスト

北川善彬，小町守：深層ニューラルネットワークを利用した日本語単語分割，言語処理学会第 21 回年次大会ワークショップ (2016)