

深層リカレントニューラルネットワークを用いた 日本語述語項構造解析

大内 啓樹^{1,a)} 進藤 裕之^{1,b)} 松本 裕治^{1,c)}

概要: 既存の日本語述語項構造解析では、形態素解析や構文解析を行い、それらの情報に基づいて各述語の項を予測していた。しかし、パイプライン処理に起因する誤り伝搬や、人手による素性設計コストの上昇などが問題点となっている。本研究では、深層リカレントニューラルネットワークを利用し、単語などの表層情報のみを用いて述語項構造解析を行う手法を提案する。また、複数の述語を同時に考慮して述語構造解析の精度を向上させるために、述語ごとのモデルをグリッド上に連結した深層リカレントグリッドモデルを提案する。NAIST テキストコーパスを用いた実験の結果、これまでに報告されている最先端の日本語述語項構造解析器の精度を上回ることを確認した。

1. はじめに

述語項構造解析は、「誰が、何を、誰に、どうした」という述語と項の関係を同定する意味解析タスクである。従来の述語項構造解析では、まずはじめに文の形態素解析や構文解析を行い、それらの情報に基づいて各述語の項を予測するというアプローチが主流であった [7], [12], [15], [20], [21], [24]。形態素や構文構造の情報が述語項構造解析の高精度化に有用であることはよく知られているが、その一方で、複数の解析モデルをパイプライン方式に連結することで生じる誤り伝搬の問題や、各サブタスクにおけるモデルの素性設計やパラメータチューニングに要するコストが大きいなどの問題がある。

本研究では、多層ニューラルネットワークの一種である深層リカレントニューラルネットワーク [16], [22] を用いて、日本語の述語項構造解析を行うモデルを提案する。提案モデルでは、単語などの表層情報のみから特徴ベクトルを計算し、その特徴ベクトルを用いて各単語が述語の項となるかどうかを判別する。近年の研究では、深層ニューラルネットワークによって、文法的・意味的な情報を反映した単語や句の特徴ベクトルを学習できることが示されており [23], [25], [26]、本研究においても、単語の表層情報から述語項構造の正解ラベルを直接予測するようにモデルを最適化することによって、明示的な品詞や構文情報がなくて

も頑健に動作することが期待できる。

より具体的には、提案モデルでは、対象となる述語が与えられたときに、文に含まれる単語列に対して双方向のリカレントニューラルネットワークを多層に重ねていき、特徴ベクトルを抽出する。そして、各単語の特徴ベクトルから、その単語が述語の項となるかどうかを判別する。多くの従来研究では、各述語ごとに項を同定するモデルを独立に構築しており、複数の述語を同時に考慮することができていないという問題がある [7], [11], [12], [21]。そこで本研究では、述語ごとのモデルを連結して新たにグリッド型のニューラルネットワークを構成し、複数の述語を同時に考慮するモデル（深層リカレントグリッドモデル）を提案する。NAIST テキストコーパスを用いた実験の結果、提案手法は、これまでに報告されている最先端の日本語述語項構造解析器 [15], [28] の精度を上回ることを確認した。

本研究で作成したプログラムコードは、著者のページにて公開する。^{*1}

2. 日本語述語項構造解析

本節では、日本語の述語項構造解析について概要を述べる。日本語述語項構造解析は、(i) 文内述語項構造解析と (ii) 文間述語項構造解析に分けられる。前者は、対象となる述語の項が文内にあるもののみを解析対象とし、後者は、対象となる述語の項が文外にあるものも解析対象に含むという違いがある。

項は、述語との位置関係により、次の3つに分類される [12], [24], [29]。

¹ 奈良先端科学技術大学院大学
Nara Institute of Science and Technology
a) ouchi.hiroki.nt6@is.naist.jp
b) shindo@is.naist.jp
c) matsu@is.naist.jp

^{*1} <https://github.com/hiroki13/neural-pasa-system>

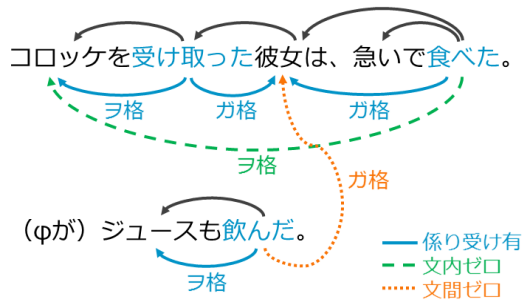


図 1 述語項構造の例.

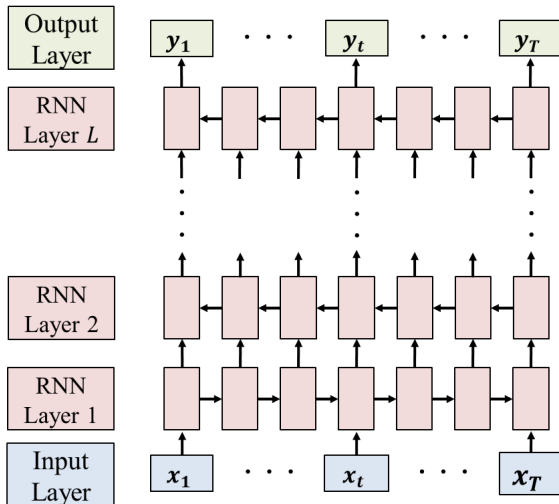


図 2 深層リカレントモデル.

係り受け有: 述語と直接係り受け関係にある項
文内ゼロ: 述語と直接係り受け関係にない文内項
文間ゼロ: 文内にはない項

図 1 は、これら 3 つの項の例を示している。述語「受け取った」と、そのガ格の項である「彼女」との間には、直接的な係り受け関係がある。このような項を「係り受け有」の項と呼ぶ。一方で、述語「食べた」と、そのヲ格の項である「クロッケー」との間には、直接的な係り受け関係がない。このような項を「文内ゼロ」と呼ぶ。

上記の 2 種類の項は、述語と同じ文内にある。しかし、述語「飲んだ」のガ格の項「彼女」は、「飲んだ」の含まれる文の中にはない。このような項を「文間ゼロ」と呼ぶ。文間ゼロを同定するには、文書全体を探索する必要があるため、文内述語項構造解析よりも効率的な探索手法が求められる [15], [20]。

本研究では、日本語の文内述語項構造解析に取り組むこととする。文間項も含めた述語項構造解析は、今後の課題とする。

3. 深層リカレントモデル

本研究では、Zhou ら [26] のモデルをベースとした深層リカレントモデル (Deep Recurrent Model; DRM) をベースラインとして用いる。図 2 は DRM を示している。DRM

<単語>	彼女	は	クロッケー	を	食べた	。
<ラベル>	GA	NONE	WO	NONE	PRED	NONE



	素性			
	Arg	Pred	Pred Context	Mark
1	彼女	食べた	を 食べた。	0
2	は	食べた	を 食べた。	0
3	クロッケー	食べた	を 食べた。	0
4	を	食べた	を 食べた。	1
5	食べた	食べた	を 食べた。	1
6	。	食べた	を 食べた。	1

図 3 素性抽出の例.

は以下のように構成されている。

入力層: 素性ベクトルからなる系列を受け取る。

中間層: 双方向型 RNN を用いる。

出力層: ソフトマックス関数を用い、多値分類を行う。

図 2 の入力層 (Input Layer) には、入力文の各単語 $\{x_t\}_1^T$ に割り当てられた素性ベクトル $\{x_t\}_1^T$ が与えられる。中間層 (RNN Layer) では、双方向型リカレントニューラルネットワーク (Bi-Directional Recurrent Neural Networks; Bi-RNN) [4], [5], [19] を用いる。出力層では、ソフトマックス関数を用いて、各単語に対してラベルを予測する。DRM は、一つの述語に対して、単語が項となるかを予測するモデルであるため、文中に複数の述語が含まれる場合には、述語ごとに別々に DRM を用意する。

例として、図 3 の上の表は、単語分割された入力文と、予測対象のラベルを示している。ターゲットとなる述語 (PRED) 「食べた」のガ格 (GA) は「彼女」であり、ヲ格 (WO) は「クロッケー」である。項とは関係のない単語には NONE ラベルが付与される。したがって、予測するラベル集合は、GA, WO, NI の 3 つの格ラベル、述語ラベル PRED, または項ではないことを表す NONE ラベルからなる。これら 5 つのラベルから、1 つのラベルを各単語に付与することが目的である。以降の節で、これらのネットワーク構成について詳しく説明する。

3.1 入力層

単語分割された入力文から、(i) 各単語にたいして素性を抽出し、(ii) 抽出された素性から素性ベクトルをつくる。

素性抽出

図 3 の下表は、各単語に対して抽出される素性の例を示している。素性として、以下の 4 種類を定義する。

Arg: 項候補の単語

Pred: 述語の単語

Pred Context: 述語のまわりの単語

Mark: Pred Context の範囲に入っているか否か

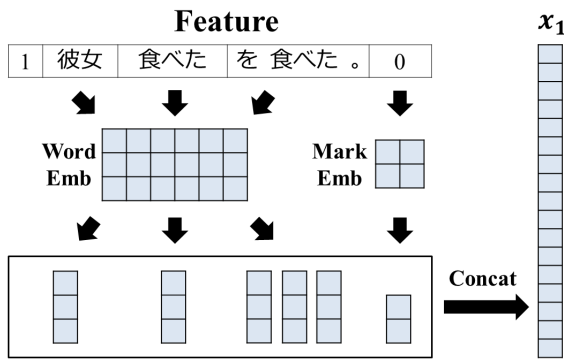


図 4 素性ベクトルの作り方の例.

Arg と Pred は当該項候補・述語の単語である。例えば、図 3 の例文において、単語「彼女」の Arg 素性は「彼女」であり、Pred 素性は「食べた」である。Pred Context 素性は、当該述語とそれに隣接する C 単語からなる素性である。 C はハイパーパラメータであり、図 3 では $C = 1$ に設定されている。例えば、単語「彼女」の Pred Context 素性は、「を」「食べた」「。」である。

Mark 素性は、Pred Context 素性に該当する単語集合に含まれるか否かを、0 または 1 で表した 2 値素性である。例として、単語「彼女」の Mark 素性を考える。Pred Context 素性は「を」「食べた」「。」の 3 つであり、「彼女」という単語は、これら 3 つの単語のどれにも該当しないため、Mark 素性は 0 である。一方、単語「を」の Mark 素性を考えると、Pred Context 素性である 3 つの単語の 1 つに該当するため、Mark 素性は 1 となる。

素性ベクトル作成

抽出した素性をもとに、素性ベクトルに作成する。図 4 は、項候補の単語である「彼女」に対して割り当てられた素性を用い、素性ベクトルを作る過程を示している。

図 4 中の「Word Emb」は、 W 個の d_w 次元単語ベクトルからなる行列 ($\mathbb{R}^{W \times d_w}$) である。単語に関連する素性である Arg・Pred・Pred Context の 3 つは、この行列の各列ベクトルに紐づいている。したがって、それぞれの単語に紐づいたベクトルを、この行列から抽出し、素性ベクトルに用いる。

同様に、図 4 中の「Mark Emb」は、 M 個の d_m 次元ベクトルからなる行列 ($\mathbb{R}^{M \times d_m}$) である。各 Mark 素性に紐づいた列ベクトルを抽出し、素性ベクトルに用いる。それぞれ抽出されたベクトル (図 4 下部の 6 本のベクトル) を、1 つに結合したものを素性ベクトル $\mathbf{x}_t$ として用いる。この素性ベクトルは、中間層の Bi-RNN の入力として与えられる。

3.2 中間層

中間層では、RNN を用いた複数の層 (RNN Layer) で再

帰的に素性ベクトルを計算する。RNN 層は、奇数番目のものは系列を左から右に、偶数番目は右から左に処理する Bi-RNN を採用する。これらの RNN 層を重ねることによって、ネットワーク構造の深さを変えることが可能である。

奇数番目の RNN 層 l では、各時刻 t において、 $l-1$ 層目の RNN の素性ベクトル $\mathbf{h}_t^{(l-1)}$ と時刻 $t-1$ での素性ベクトル $\mathbf{h}_{t-1}^{(l)}$ を入力とし、素性ベクトル $\mathbf{h}_t^{(l)}$ を計算する。

$$\mathbf{h}_t^{(l)} = f(\mathbf{h}_t^{(l-1)}, \mathbf{h}_{t-1}^{(l)}) \quad (1)$$

偶数番目の RNN 層では、右から左に伝搬するため、 $\mathbf{h}_{t-1}^{(l)}$ の代わりに $\mathbf{h}_{t+1}^{(l)}$ が用いられる。

一般的な RNN において、関数 $f(\cdot)$ は次のように定義される。

$$f(\mathbf{x}_t, \mathbf{h}_{t-1}) = g(\mathbf{W}_x \mathbf{x}_t + \mathbf{W}_h \mathbf{h}_{t-1})$$

$\mathbf{x}_t$ は時刻 t における入力ベクトルであり、 $\mathbf{h}_{t-1}$ は時刻 $t-1$ における素性ベクトルを表す。関数 $g(\cdot)$ は任意の非線形関数である*2。このような RNN は、勾配の消失・爆発などの問題から、学習が困難であることが指摘されている [17]。これらの問題を回避し、系列の長期的な記憶の保持を目的として Long Short-Term Memory (LSTM) [9] や Gated Recurrent Unit (GRU) [2] が提案されている。本研究では、関数 $f(\cdot)$ として GRU を用いる。

3.3 出力層

出力層では、ソフトマックス関数を用いて、多値分類を行う。

$$\mathbf{y}_t = \text{softmax}(\mathbf{W}_y \mathbf{x}_t^{(L)})$$

$\mathbf{x}_t^{(L)}$ は、最後の RNN 層から伝搬されるベクトルである (図 2)。 $\mathbf{y}_t$ の各要素は、各ラベルに対応する確率値である。この中から、確率最大のラベルを解析結果として出力する。ラベルの種類数は GA, O, NI, PRED, NONE の 5 つである。

Zhou ら [26] のモデルでは、出力層に一次の条件付確率場 (Conditional Random Fields; CRF) を用いている。しかし、我々の行った日本語述語項構造解析の予備実験では、CRF による精度向上は確認できなかった。この理由として、単語単位で日本語の述語項構造解析を行う場合、ガ・ヲ・ニ格のラベルが隣り合う事例は少ないことが考えられる。例えば、図 3 の事例では、「彼女」と「コロッケ」にそれぞれガ格・ヲ格のラベルが付与されているが、それらの間に格助詞「は」が存在する。したがって、一次の CRF では、ガ格・ヲ格間の依存関係を直接考慮することが難しい。本研究では、CRF ではなく、より計算量の少ないソフトマックス関数を出力層でのラベル推定に用いることにする。

*2 本研究では $g(\cdot)$ として $\tanh(\cdot)$ を用いる。

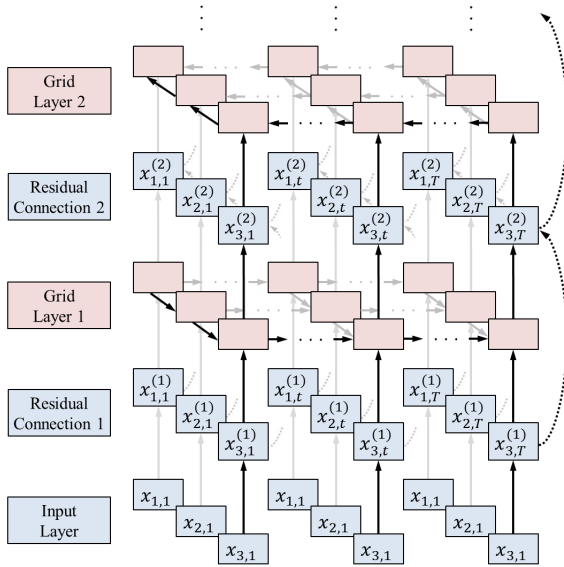


図 5 深層リカレントグリッドモデル: 3 本の系列の例.

4. 深層リカレントグリッドモデル

前節の深層リカレントモデル (DRM) では、1 つの述語に対する項候補の単語系列 $\{x_t\}_1^T$ から素性ベクトルの系列 $\{\mathbf{x}_t\}_1^T$ を作り、それをネットワークの入力として用いていた (第 3 章). したがって、DRM では、複数の述語間の関係を考慮することができないという問題がある [15], [24].

そこで本研究では、DRM を拡張し、深層リカレントグリッドモデル (Deep Recurrent Grid Model; DRGM) を提案する. DRGM は、述語ごとの DRM を連結してグリッド型のネットワークを構成し、複数の述語間の関係を同時に考慮して項の推定を行うモデルである. 図 5 に、DRGM の概要を示す. DRGM の特徴は以下の点である.

入力: P 個の述語に対して作られた P 本の素性ベクトルの系列を受け取る.

系列をまたぐ接続: グリッド型のネットワーク構造により、異なる系列の素性ベクトルも利用できる.

層をまたぐ接続: Residual Connections を利用することにより、層をまたぐ素性ベクトルを利用できる.

以降の節で、これらの特徴について詳しく説明する.

4.1 複数の述語とその項候補からなる入力

DRGM は、述語の数 P と同じ本数の系列を入力とする. 図 6 は、3 つの述語を含む 12 単語からなる 1 文から、入力ベクトルが作られるまでの流れを示している. 例文において、「起きて」「食べ」「向かった」の 3 つが述語として与えられている. DRM と同様、それぞれの述語に対し、予測対象となる項候補からなる系列を構成し、各項候補に対して素性ベクトルを割り当てる. この処理を、各述語に対応する系列のそれぞれに対して行う. 上記の例では、結果と



図 6 深層リカレントグリッドモデルの入力ベクトル作成例.

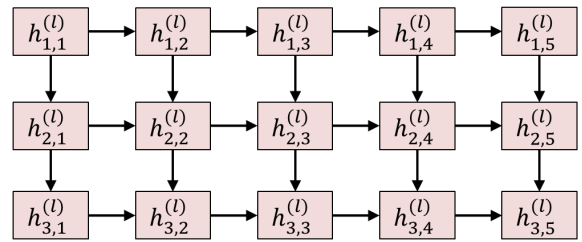


図 7 グリッド層: 奇数層の例.

して 3 行 12 列、合計 36 本の素性ベクトルを得る.

言い換えれば、

- (1) P 個の述語を含む T 単語からなる文 $\mathbf{x} = \{x_t\}_1^T$ が与えられる.
- (2) 各述語 p に対する項候補の単語 $x_{p,t}$ に素性ベクトルを作り、素性ベクトルの系列 $\mathbf{x}_p = \{\mathbf{x}_{p,t}\}_1^T$ を得る.
- (3) (2) の結果、 P 本の素性ベクトルの系列 $\{\mathbf{x}_p\}_1^P$ を得る. このようにして得られた複数の素性ベクトルの系列 $\{\mathbf{x}_p\}_1^P$ を、ネットワークの入力として用いる.

4.2 グリッドネットワーク

図 7 は、図 5 内のグリッド層の奇数層 1 層のみを取り出したものである. 奇数層では、左上から右下に向かって、素性ベクトルを再帰的に計算する.

定式的には、奇数層 l において、 p 番目の系列の t 番目の素性ベクトル $\mathbf{h}_{p,t}^{(l)}$ は、次の式にしたがって計算される:

$$\mathbf{h}_{p,t}^{(l)} = f([\mathbf{x}_{p,t}^{(l)}, \mathbf{h}_{p-1,t}^{(l)}, \mathbf{h}_{p,t-1}^{(l)}]) \quad (2)$$

ここで、 $\mathbf{x}_{p,t}^{(l)}$ は、 l 層目の p 番目の系列の t 番目の入力ベクトルを表す. $\mathbf{h}_{p-1,t}^{(l)}$ は、 l 層目の $p-1$ 番目の系列の t 番目の素性ベクトルを表す. これら 2 つのベクトルを結合し、関数 $f(\cdot)$ の入力として用いる. 同様に、 $\mathbf{h}_{p,t-1}^{(l)}$ は l 層目の p 番目の系列の $t-1$ 番目の素性ベクトルを表し、関数 $f(\cdot)$ の入力として用いられる. DRM と同様、関数 $f(\cdot)$ として GRU を用いる.

偶数層でも同様の計算が行われるが、奇数層とは逆で、右下から左上に向かって再帰計算が行われる. 出力層は DRM と同様、ソフトマックス関数を用いる.

4.3 Residual Connections

層を重ねて深いネットワークを構成する場合、勾配の爆発・消失などの問題により、ネットワークの学習が上手くできない場合が多い。そこで本研究では、各層間の接続に Residual Connections [8] を導入する。Residual Connections は、深いネットワークの効果的な学習に寄与するとの報告がある [8], [23]。

図5において、各「Residual Connection」層が点線つながれているように、 l 層目のグリッド層の入力として、 $l-1$ 層目の出力ベクトルだけでなく、 $l-1$ 層目の入力ベクトルも用いられる。具体的には、式2における入力ベクトル $\mathbf{x}_{p,t}^{(l)}$ は、 $l-1$ 層目の RNN の入力ベクトル $\mathbf{x}_{p,t}^{(l-1)}$ と出力ベクトル $\mathbf{h}_{p,t}^{(l-1)}$ を加算したベクトルである。

$$\mathbf{x}_{p,t}^{(l)} = \mathbf{x}_{p,t}^{(l-1)} + \mathbf{h}_{p,t}^{(l-1)}$$

ただし、1層目の入力ベクトル $\mathbf{x}_{p,t}^{(1)}$ は、素性ベクトル $\mathbf{x}_{p,t}$ と重みパラメータ $\mathbf{W}_x$ の内積を計算によって計算される。

Residual Connections は、DRM にも適用可能である。したがって、実験では両モデルにおいて Residual Connections の有無による性能の違いも確認する (第5章)。

5. 実験

本研究では、単語単位の日本語述語項構造解析を行う。

5.1 実験設定

データとして、NAIST テキストコーパス 1.5 [10] を用いる。実験では、次に示すような、標準的なデータ分割法を採用し、モデルの訓練・開発・評価を行った [21]。

訓練: 1月1-11日の記事と、1月から8月の社説。

開発: 1月12, 13日の記事と、9月の社説。

評価: 1月14-17日の記事と、10月から12月の社説。

実験で用いる単語境界として、NAIST テキストコーパスにアノテーションされている単語境界を利用する。モデルの学習に、外部資源は一切利用しない。

5.2 実装詳細

本研究の提案モデルの実装は、深層学習ライブラリ Theano [1] を利用し、CPU(Intel 6 Core Xeon E5-4617)上で実行する。Pred Context 素性(3節)の C は2に設定する。使用する単語ベクトルとして、訓練データ内に出現する単語の中で、頻度2以上のもののみを使用し、残りの単語は未知語として未知語のベクトルに割り当てる。エポック数は50に設定し、開発データのF値が最も良いエポックでの評価データの結果を報告する。パラメータの最適化は、ミニバッチを利用した確率的勾配降下法 (SGD) で行う。ミニバッチサイズは、DRM は8, DRGM は4に設定する。学習係数はAdam [14] を用いて自動調整する。各ハ

イパーパラメータは、以下のように選ぶ。

単語ベクトル : 32次元に設定し、初期値として $[-0.01, 0.01]$ から一様分布に従ってサンプリングした値を設定する。

重みパラメータ行列 : 32×32 とし、 $[-0.01, 0.01]$ から一様分布に従ってサンプリングした値を設定する。

正則化項 : 正則化項のハイパーパラメータ λ は $[0.001, 0.0005, 0.0001]$ の中から、開発データのF値が最大となるものを選ぶ。

5.3 比較手法

先行研究では、それぞれ実験設定が異なっており、厳密な比較はできないが、参考のため、NAIST テキストコーパス 1.5 を用いた先行研究との比較を行う。日本語述語項構造解析手法は、点推定 (Pointwise) アプローチと同時推定 (Joint) アプローチに分けられるため、それらの手法と比較する。

点推定のアプローチとして、Imamura ら [12] の手法を比較対象とする*3。この手法は、1つの述語に対してそれぞれの項候補のスコアを点推定し、各述語に対してスコア最大の項候補を選ぶことによって、述語項構造を求める。

同時推定のアプローチとして、Ouchi ら [15] の全格同時解析モデル (ALL-Cases Joint Model) と比較する。この手法は、文内の全述語項間の関係を考慮し、最適な組み合わせを近似的に求める。*4

また、効果的な素性選択に基づいて点推定を行い、高精度な解析結果を残している松林ら [28] の手法も比較対象とする*5。ただし、松林らの手法では、京都大学格フレーム [13] などの外部資源を使用している点が、他の比較手法と異なる。

5.4 結果

表1に、本研究の提案手法と比較手法のF値を示す。「中間層」の列は、各モデルにおける中間層 (RNN 層, グリッド層) の数を表す。

全体の性能

開発データのF値が最高を記録した層数における、評価データの精度を比較する。DRM は層数8のときにF値80.70%を記録している。一方、DRGM は層数6のときにF値81.22%であり、DRM を約0.5ポイント上回った。この結果から、複数の述語項を考慮することにより、性能の向上が可能となることが示唆された。

*3 同データセットを使った実験結果として、Ouchi ら [12] で用いられた再実装の結果と比較する。

*4 Imamura らと Ouchi らの手法は、文節単位で評価している。

*5 松林らの合計のF値 (図1) には、文節内の項の解析結果も含まれている。

	中間層	開発	評価
DRM	2	80.14	80.58
	4	80.66	80.91
	6	80.52	80.37
	8	80.74	80.70
DRGM	2	80.43	80.63
	4	80.78	80.93
	6	81.14	81.22
	8	80.90	81.06
IM09	-	-	78.15
MA14*	-	-	80.9
OU15	-	-	79.23

表 1 述語項構造解析結果の F 値。IM09 は Imamura et al. (2009) を、MA14*は松林ら (2014) を、OU15 は Ouchi et al. (2015) を表している。MA14*(松林ら) は外部資源を用いている。

	中間層	有	無
DRM	2	80.14	80.44
	4	80.66	80.45
	6	80.52	80.35
	8	80.74	80.70
DRGM	2	80.43	80.44
	4	80.78	80.67
	6	81.14	80.75
	8	80.90	80.61

表 2 Residual Connections の利用の有無による F 値。

既存手法との比較

点推定手法である Imamura ら (IM09) の F 値 78.15%を、提案モデルが 2.5-3.0 ポイント上回っている。同様に、同時推定手法である Ouchi ら (OU15) の F 値 79.23%をも、1.5-2.0 ポイント上回る結果となっている。また、外部資源を用いた松林ら (MA14) の F 値 80.9 も、外部資源を用いない DRGM がわずかながら上回っている。

中間層の深さの影響

DRM の F 値は、評価データでは中間層数 4 で最高を記録し、それ以降は頭打ちとなっている。一方、DRGM の F 値は、中間層数 6 までは順調に上昇し、中間層数 8 で低下している。これらの結果から、中間層は適切な数に設定することで良い性能が期待できるが、浅すぎたり深すぎたりすると性能が低下する傾向にあることが示唆された。

Residual Connections の効果

図 2 は、Residual Connections を利用した場合としない場合の、開発データにおける F 値を示している。DRM では、Residual Connections を利用してもしなくても、結果に大きな差は見られない。一方、DRGM では、中間層が 2 層の場合は同程度の結果であるが、4-8 層の場合に 0.3-0.4 ポイントの差が生じている。この結果は、Residual Connections

	係り受け有			文内ゼロ		
	ガ	ヲ	ニ	ガ	ヲ	ニ
DRM	88.74	92.90	64.84	51.24	35.13	8.90
DRGM	88.66	93.95	66.50	51.57	38.06	9.44
IM09	86.50	92.84	30.97	45.56	21.38	0.83
MA14	87.8	94.0	63.7	49.0	27.7	25.7
OU15	88.13	92.74	38.39	48.11	24.43	4.80

表 3 格ごとの F 値の比較。

の利用が、モデルの学習に一定の効果があったことを示している。しかし、両モデル間で効果が異なる理由は明らかでないため、今後詳細な調査を行っていききたい。

格ごとの性能

表 3 は、格ごとの F 値を表している。提案手法では、ガ・ヲ格の文内ゼロの F 値が大きく改善している。

これまで、文内ゼロの解析には、長距離の依存関係をとらえる必要があるため、局所的な素性のみを用いる手法では解析が困難であった [18], [27], [30]。一方、DRM と DRGM は、文内のすべての単語を考慮し、かつ、それらの長距離の依存関係をとらえることができるため、解析結果の向上につながったと考えられる。

6. 関連研究

6.1 日本語述語項構造解析

既存の日本語述語項構造解析手法は、大きく 2 つのカテゴリに分けられる。

- 点推定手法
- 同時推定手法

点推定手法では、1 つの述語に対してそれぞれの項候補のスコアを推定し、スコア最大の項候補を項として選出する [7], [11], [12], [21]。同時推定では、1 文内の複数の述語項を考慮したスコアを推定し、複数の項を同時に選出する [6], [15], [18], [20], [24]。

本研究の提案モデルは、素性として文内の述語間の関係を考慮しているが、各述語に対するそれぞれの項の推定は独立に行っているため、点推定のカテゴリに属す。

6.2 ニューラルネットを用いた日本語述語項構造解析

近年、日本語述語項構造解析に、ニューラルネットワークを適用する試みがなされている [11], [20]。Shibata らは、Ouchi らのモデルのスコア計算部分にフィードフォワードニューラルネットワークを使用している。Iida らは、Multi-column Convolutional Neural Network (MCNN) を用いて、ガ格の文内ゼロの同定を行っている。いずれの手法も、単語情報に加え、品詞や構文情報をニューラルネットワークの素性として使用している。ニューラルネットワークの使用によって、それらの組み合わせ素性を人手で

設計せずに、高精度な解析を実現している。

本研究では、品詞や構文情報なども使用しないため、さらなる素性設計コストの削減に貢献している。また、それらの先行研究は、それぞれ異なる実験設定で性能を評価しているため比較はできないが、今後、実験設定を揃えた比較実験を行いたい。

6.3 ニューラルネットを用いた英語述語項構造解析

品詞や構文情報を用いず、単語情報のみから述語項構造(意味役割)を予測する手法が提案されている。Collobertらの手法[3]では、畳込みニューラルネットワーク(Convolutional Neural Network; CNN)を用いて、句構造に基づいた英語述語項構造解析タスク(CoNLL-2005 Shared Task)に取り組み、F値で74.15%の解析精度を達成している。同様のタスクにおいて、Zhouら[26]は、多層リカレントニューラルネットワークと条件付確率場(Conditional Random Fields; CRF)を利用することにより、これまでの最高精度にあたるF値81.07%を達成した。

本研究のDRMは、Zhouらのモデルの拡張として、各層間を結ぶResidual Connectionsを導入し、CRF部分をソフトマックス関数に置き換えた。また、Zhouらのモデルでは、1本の系列に基づいて項を推定していたが、本研究のDRGMは、複数系列を考慮することを可能とした。

7. おわりに

本研究では、日本語述語項構造解析のために、深層リカレントニューラルネットワークを利用したモデルを提案した。実験の結果、単語などの表層情報のみから、高精度に解析が行えることが明らかになった。特に、複数の述語を考慮した深層リカレントグリッドモデルが良い性能を示した。このことから、文内の述語間の関係が項構造同定の手がかりになるというYoshikawaら[24]やOuchiら[15]の報告が、リカレントニューラルネットを用いた場合にも成り立つことが示された。

本研究の実験結果から、Residual Connectionsが効果を発揮するモデルとそうでないモデルがあることが示されたが、その要因は明らかではないため、より詳細な分析が必要である。また、提案モデルにより文内ゼロの解析精度が改善したが、まだ解析に失敗している事例も多々あり、成功した事例と失敗した事例の分析と、それを踏まえたモデルの改善を行っていく予定である。

謝辞 東北大学の松林優一郎氏と、情報通信研究機構の飯田龍氏にさまざまなご教示を頂いたことを深謝する。また、奈良先端科学技術大学院大学の椿真史氏、吉野幸一郎氏、意味談話解析勉強会メンバーにも、数々の助言や添削をしていただいたことに感謝する。

参考文献

- [1] Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I. J., Bergeron, A., Bouchard, N. and Bengio, Y.: Theano: new features and speed improvements, Deep Learning and Unsupervised Feature Learning NIPS 2012 Workshop (2012).
- [2] Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. and Bengio, Y.: Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation, *Proceedings of EMNLP*, pp. 1724–1734 (2014).
- [3] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K. and Kuksa, P.: Natural Language Processing (Almost) from Scratch, *Journal of Machine Learning Research* (2011).
- [4] Graves, A., Jaitly, N. and Mohamed, A.-r.: Hybrid speech recognition with deep bidirectional LSTM, *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop* (2013).
- [5] Graves, A., Fernández, S. and Schmidhuber, J.: Bidirectional LSTM networks for improved phoneme classification and recognition, *International Conference on Artificial Neural Networks*, pp. 799–804 (2005).
- [6] Hangyo, M., Kawahara, D. and Kurohashi, S.: Japanese Zero Reference Resolution Considering Exophora and Author/Reader Mentions, *Proceedings of EMNLP*, pp. 924–934 (2013).
- [7] Hayashibe, Y., Komachi, M. and Matsumoto, Y.: Japanese Predicate Argument Structure Analysis Exploiting Argument Position and Type, *Proceedings of IJCNLP*, pp. 201–209 (2011).
- [8] He, K., Zhang, X., Ren, S. and Sun, J.: Deep residual learning for image recognition, *arXiv preprint arXiv:1512.03385* (2015).
- [9] Hochreiter, S. and Schmidhuber, J.: Long Short-Term Memory, *Neural Computation* (2013).
- [10] Iida, R., Komachi, M., Inui, K. and Matsumoto, Y.: Annotating a Japanese text corpus with predicate-argument and coreference relations, *Proceedings of the Linguistic Annotation Workshop*, pp. 132–139 (2007).
- [11] Iida, R., Torisawa, K., Oh, J.-H., Kruengkrai, C. and Kloetzer, J.: Intra-Sentential Subject Zero Anaphora Resolution using Multi-Column Convolutional Neural Network, *Proceedings of EMNLP*, pp. 1244–1254 (2016).
- [12] Imamura, K., Saito, K. and Izumi, T.: Discriminative Approach to Predicate-Argument Structure Analysis with Zero-Anaphora Resolution, *Proceedings of ACL-IJCNLP*, pp. 85–88 (2009).
- [13] Kawahara, D. and Kurohashi, S.: Case frame compilation from the web using high-performance computing, *Proceedings of LREC*, pp. 1344–1347 (2006).
- [14] Kingma, D. and Ba, J.: Adam: A Method for Stochastic Optimization, *arXiv preprint arXiv: 1412.6980* (2014).
- [15] Ouchi, H., Shindo, H., Duh, K. and Matsumoto, Y.: Joint Case Argument Identification for Japanese Predicate Argument Structure Analysis, *Proceedings of ACL-IJCNLP*, pp. 961–970 (2015).
- [16] Pascanu, R., Gulcehre, C., Cho, K. and Bengio, Y.: How to construct deep recurrent neural networks, *arXiv preprint arXiv:1312.6026* (2013).
- [17] Pascanu, R., Mikolov, T. and Bengio, Y.: On the difficulty of training recurrent neural networks, *Proceedings of ICML* (2013).
- [18] Sasano, R. and Kurohashi, S.: A Discriminative Approach to Japanese Zero Anaphora Resolution with Large-scale Lexicalized Case Frames, *Proceedings of IJCNLP*, pp. 758–766 (2011).
- [19] Schuster, M. and Paliwal, K. K.: Bidirectional recurrent neural networks, *IEEE Transactions on Signal Processing*, pp. 2673–2681 (1997).
- [20] Shibata, T., Kawahara, D. and Kurohashi, S.: Neural Network-Based Model for Japanese Predicate Argument Structure Analysis, *Proceedings of ACL*, pp. 1235–1244 (2016).
- [21] Taira, H., Fujita, S. and Nagata, M.: A Japanese Predicate Argument Structure Analysis using Decision Lists, *Proceedings of EMNLP*, pp. 523–532 (2008).
- [22] Wang, Y. and Tian, F.: Recurrent Residual Learning for Sequence Classification, *Proceedings of EMNLP*, pp. 938–943 (2016).
- [23] Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K. et al.: Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation, *arXiv preprint arXiv:1609.08144* (2016).
- [24] Yoshikawa, K., Asahara, M. and Matsumoto, Y.: Jointly Extracting Japanese Predicate-Argument Relation with Markov Logic, *Proceedings of IJCNLP*, pp. 1125–1133 (2011).
- [25] Zhou, J., Cao, Y., Wang, X., Li, P. and Xu, W.: Deep Recurrent Models with Fast-Forward Connections for Neural Machine Translation, *Transactions of the Association for Computational Linguistics*, pp. 371–383 (2016).
- [26] Zhou, J. and Xu, W.: End-to-end learning of semantic role labeling using recurrent neural networks, *Proceedings of ACL-IJCNLP* (2015).
- [27] 松林優一郎, 中山周, 乾健太郎: 日本語述語項構造解析タスクにおける項の省略を伴う事例の分析, *自然言語処理*, pp. 433–463 (2015).
- [28] 松林優一郎, 乾健太郎: 統計的日本語述語項構造解析のための素性設計再考, *言語処理学会第 20 回年次大会発表論文集*, pp. 360–363 (2014).
- [29] 林部祐太, 小町守, 松本裕治: 述語と項の位置関係ごとの候補比較による日本語述語項構造解析, *自然言語処理*, pp. 3–25 (2014).
- [30] 飯田龍, 乾健太郎, 松本裕治: 文脈の手がかりを考慮した機械学習による日本語ゼロ代名詞の先行詞同定, *情報処理学会論文誌*, pp. 906–918 (2004).