

ニューラルネットワークを利用した日本語意味役割付与モデルの構築

岡村 拓哉^{1,a)} 竹内 孔一^{1,b)} 石原 靖弘^{2,c)}

受付日 2019年1月29日, 採録日 2019年8月9日

概要: 本論文では日本語の意味役割ラベルを決定する有効なモデルを明らかにするために, 日本語均衡コーパスに 62 種類の意味役割が付与されている BCCWJ-PT コーパスに対してニューラルネットワークを利用した日本語意味役割付与モデルの構築について記述する. ニューラルネットワークとして 3 層ニューラルネットワーク, 畳込みニューラルネットワーク, 再帰型ニューラルネットワークの一種である GRU, 双方向 GRU に max-pooling を適用したモデルを利用する. 実験結果から, 畳込みニューラルネットワーク, 双方向 GRU モデルが従来の機械学習モデルの 1 つである SVM に対して高い識別精度が得られたことを示す. 次に, 特徴量の選択や転移学習の利用により 3 層ニューラルネットワーク, および, GRU モデルで精度の向上が見られたことを報告する. さらに双方向 GRU モデルが最も高い精度が示すことを報告する.

キーワード: 意味役割付与, 述語項構造解析, 双方向 GRU モデル, max-pooling

Using Neural Networks to Construct a Japanese Semantic Role Labeling Model

TAKUYA OKAMURA^{1,a)} KOICHI TAKEUCHI^{1,b)} YASUHIRO ISHIHARA^{2,c)}

Received: January 29, 2019, Accepted: August 9, 2019

Abstract: We discuss effective features and learning methods of neural network models for deciding semantic role labels in Japanese. The features and models were evaluated using BCCWJ-PT, which is a Japanese corpus annotated with 62 types of semantic role labels. We applied several types of neural network models, such as a simple three-layer model, convolutional neural network model, GRU model, and bi-directional GRU with max-pooling model. The experimental results show that: 1) The SVM based model was outperformed by the convolutional neural network model and bi-directional GRU with max-pooling model. and 2) Grammatical features and transfer learning were only effective for the three-layer and GRU models. The bi-directional GRU with max-pooling model had the best performance among the neural network models.

Keywords: semantic role labeling, predicate argument structure analysis, bi-directional GRU model, max-pooling

1. はじめに

本論文ではニューラルネットワークを利用した日本語意味役割付与モデルの構築について記述する. 述語の係り元 (本論文では項と呼ぶ) である主語や目的語に対して意味的な関係をラベル付けする意味役割付与タスクが英語を中心に発展してきており, 初期の研究では意味役割を決定する統語的および文法的な特徴を利用したモデルが提案され [8], 近年では深層学習を利用した手法が提案されてい

¹ 岡山大学大学院自然科学研究科
Graduate School of Natural Science and Technology,
Okayama University, Okayama 700-8530, Japan

² 岡山大学工学部
Faculty of Engineering, Okayama University, Okayama 700-0082, Japan

a) t-okamura@s.okayama-u.ac.jp

b) takeuc-k@okayama-u.ac.jp

c) ishiharay@okayama-u.ac.jp

る [10], [11], [22], [24], [28], [29], [31]. これらの発展の原因は主に英語における大規模意味役割付与コーパスが複数利用可能であることである*1.

日本語においては京都大学テキストコーパス [12], NAIST テキストコーパス [39] を対象に 3 種類の意味的關係 (ガ格, ヲ格, ニ格) が付与され, 照応解析を含めた述語項構造解析として近年, 様々な手法が提案されている [18], [20], [21], [27]. 一方, 本論文では, たとえば CoNLL2005 [2] の意味役割ラベルにあるように「方向 (AM-DIR)」や「時間 (AM-TMP)」といった付加詞まで含めた意味役割ラベルの同定に関する研究に注目する.

日本語の意味役割付与データとして「時間」や「方向」といった付加詞まで含めた意味役割 (62 種類) を日本語書き言葉均衡コーパス BCCWJ [17] に付与した BCCWJ-PT [34]*2 が公開されている. そこで, 本論文では BCCWJ-PT を対象に意味役割を決定するために必要な特徴量とニューラルネットワークの構造について論じる.

先行研究として, BCCWJ-PT を利用して, 項の末尾表現を統計的学習モデルの特徴量として取り込んで意味役割付与精度の向上を行った石原らの研究 [38] がある. しかしながら近年のニューラルネットワークを適用することでより精度が向上する手法が構築できる可能性がある.

本論文ではニューラルネットワークとして 3 層ニューラルネットワーク, 畳込みニューラルネットワーク, 再帰型ニューラルネットワーク (RNN) の一種である GRU, さらに双方向 GRU に max-pooling を適用した各モデルを利用する. さらに意味役割を含む広範囲な意味的關係を付与している GDA コーパス [35] を利用して転移学習 [32] による識別精度の向上を試みる.

評価実験から (1) 畳込みニューラルネットワークおよび双方向 GRU を利用した意味役割付与モデルが SVM の識別精度を上回ったこと, (2) 双方向 GRU モデルが最も高い識別精度を示したこと, (3) 先行研究での特徴量 [38] の効果は限定的であること, (4) 転移学習の適用により収束時間が早くなること, 学習データが減少した場合に識別精度の低下が和らげられること, 識別精度の向上は限定的であることを明らかにする.

2. 意味役割付与タスク

本論文で扱う意味役割付与とは文の中である述語に対する係り元の項に対して意味的關係を付与する作業である. たとえば図 1 に「太郎が詰め将棋の本を買った」の例を示す. 図 1 では, 述語「買った」に対する係り元である項は「太郎が」と「詰め将棋の本を」の 2 つがある. この

[動作主 太郎が] [対象 詰め将棋の本を] [述語 買った]

図 1 「買った」に対する意味役割付与例

Fig. 1 An example of semantic role labels for the predicate “買う (buy)”.

表 1 意味役割付与データの例

Table 1 Examples of the annotated data of semantic role labels.

意味役割ラベル	述語	項
対象	買う 買った	詰め将棋の本を
動作主	買う 買った	太郎は

ときに前者の項に「動作主」(購入者), 後者に「対象」(買われた物) という意味的關係のラベルを付与するのが本研究の意味役割付与である.

意味役割に相当する意味的關係を付与したコーパスは複数ある. 前述の京都大学コーパス, NAIST テキストコーパス以外に, 日英対訳関係を考慮した EDR コーパスでは 23 種類の意味役割が付与されている*3. また, テキストの意味的構造化に主眼をおいた GDA では約 100 種類の意味的關係が付与されている*4. 本研究では国立国語研究所が作成した日本語書き言葉均衡コーパス BCCWJ に述語項構造シソーラス (PT) の意味役割を付与した BCCWJ-PT を対象とする. 理由は均衡コーパスであるためジャンルが広いことから, 語義の事例が広いことが期待できること*5, 意味役割の体系が辞書として事例付きで公開されていることである*6.

BCCWJ-PT では BCCWJ で付与されている長単位の形態素に対して PT で定義されている述語と係り元の項の意味役割が付与されている. 本研究では項と述語の情報だけでどの程度意味役割ラベルが付与できるかを明らかにするために, BCCWJ-PT から項と述語の組を 1 事例として取り出したデータを意味役割付与データとする. たとえば上記の「太郎が詰め将棋の本を買った」の場合は表 1 に示すように, 2 つの意味役割事例に分解する. 述語の部分では表層形と基本形のみを利用し, 助動詞などは取り入れていない. 4 章の意味役割付与実験では表 1 の 1 行を 1 事例として学習および評価に利用する. 意味役割付与データに含まれる意味役割の種類数と事例数は 4 章に記述する.

3. 意味役割付与モデル

本研究で利用するニューラルネットワークを利用した意味役割付与モデルについて記述する. 複数の異なる構造のニューラルネットワークを利用するが, どの場合でも最終出力層で意味役割ラベルの識別結果を出力するのは同様で

*1 たとえば CoNLL2005 [2], 2009 [9], 2012 [26] などの評価型ワークショップのデータおよび PropBank [14] や FrameNet [1].

*2 国立国語研究所の中納言のサイト (<https://bccwj-data.ninjal.ac.jp/mdl/>) から利用可能.

*3 http://www2.nict.go.jp/ipp/EDR/JPN/J_indexTop.html

*4 <http://i-content.org/gda/tagman.html>

*5 たとえば「買う」ならば購入以外に「反発を買う」, 「けんかを買う」などが登録されている.

*6 <http://pth.cl.cs.okayama-u.ac.jp/>

ある．そこで，最終出力層に関するモデル化を行った後，各ニューラルネットワークの定義を以降の節で記述する．

本研究における意味役割付与モデルは表 1 の 1 事例のデータに対して意味役割ラベルを 1 つ推定する．そこで，項と述語の情報を K_x 次元でベクトル化した入力 $\mathbf{x} \in \mathbb{R}^{K_x}$ に対して，最終出力層のユニット j の出力を y_j とすると softmax 関数 [33] を適用することで y_j は確率となる．よって推定した意味役割ラベル $\hat{S}$ は $y_j = p(S_j|\mathbf{x})$ とすると下記の式で求められる．

$$\hat{S} = \underset{j \in \text{Sem}}{\operatorname{argmax}} p(S_j|\mathbf{x}) \quad (1)$$

ここで S_j ($j = 1, \dots, \text{Sem}$) はユニット j に対応した意味役割ラベルであり， Sem は意味役割の種類数を示す．

下記の各モデルで $\mathbf{y} = {}^T[y_1, \dots, y_j, \dots, y_{\text{Sem}}]$ を定義することで構造の異なるニューラルネットワークを意味役割付与モデルとして利用する．ここで， ${}^T[\cdot]$ は横ベクトル $[\cdot]$ を転置した列ベクトルを表す*7．

3.1 3層ニューラルネットワークモデル (3LNN)

3層順伝播型ニューラルネットワークによる出力 $\mathbf{y}$ を下記のように定義する．入力側の重みを $\mathbf{W}_{in} \in \mathbb{R}^{m \times K_x}$ とし，出力層の重みを $\mathbf{W}_{out} \in \mathbb{R}^{\text{Sem} \times m}$ とすると $\mathbf{y}$ は下記のように定義できる．

$$\mathbf{y} = f_{out}(\mathbf{W}_{out}\mathbf{h} + \mathbf{b}_{out}) \quad (2)$$

$$\mathbf{h} = f_{in}(\mathbf{W}_{in}\mathbf{x} + \mathbf{b}_{in}) \quad (3)$$

ここで $\mathbf{h} \in \mathbb{R}^m$ は中間層の出力を表す．活性化関数として中間層の出力 (f_{in}) では ReLU [19] を適用し，出力層 (f_{out}) では softmax 関数を適用する．また， $\mathbf{b}_{out} \in \mathbb{R}^{\text{Sem}}$ と $\mathbf{b}_{in} \in \mathbb{R}^m$ はそれぞれバイアス項を示す．

入力 $\mathbf{x}$ として項と述語に関する特徴量を複数組み合わせたベクトルを仮定し，精度がどのように変わるのか実験する．組合せは 4.3 節で説明する．また，ドロップアウト [15] を適用し，誤差関数として交差エントロピー誤差を使用し，学習には Adam を用いる．各パラメータの詳細は 4.5 節に記述する．

3.2 畳み込みニューラルモデル (CNN)

結合重みの自由度を減らしてから学習を行う畳み込みニューラルネットワーク (CNN) [16], [32] は主に画像処理の分野に用いられることが多いが，図 2 のように単語の分散表現を時系列順に並べ*8二次元の形状にすることにより自然言語処理に適用することが可能となる [13]．入力文を形態素に分割し， t 番目の形態素の d 次元の分散表現ベクトルを $\mathbf{x}^t \in \mathbb{R}^d$ とする．このとき， t 番目の形態素から

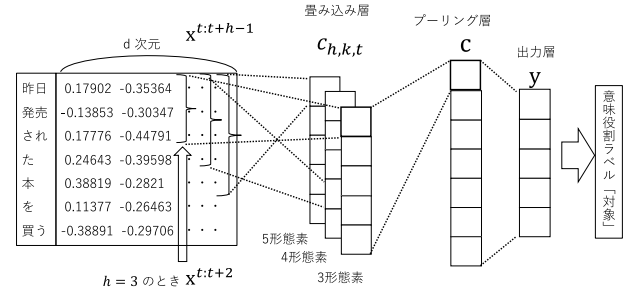


図 2 CNN による意味役割付与モデル

Fig. 2 An annotation model of semantic role labels using CNNs.

h 個の形態素の単語ベクトルを結合したベクトル $\mathbf{x}^{t:t+h-1}$ を下記のように定義する．

$$\mathbf{x}^{t:t+h-1} = \mathbf{x}^t \oplus \mathbf{x}^{t+1} \oplus \dots \oplus \mathbf{x}^{t+h-1} \quad (4)$$

ここで， $\oplus$ はベクトルどうしを結合する演算を表す．また，形態素数 h は h_{min} から h_{max} まで変化させる．

畳み込み層では h 個の形態素に対して d 次元のフィルタと呼ばれる重み $\mathbf{w}_{h,k} \in \mathbb{R}^{hd}$ を定義する*9． $k = 1, \dots, K$ は各 $\mathbf{w}_h$ に対して K 個の異なるフィルタを用意すること表しており，複数のフィルタで様々な特徴量を学習できるようにする．

各形態素 t 番目の結合ベクトル $\mathbf{x}^{t:t+h-1}$ に対してフィルタを適用した特徴量 $c_{h,k,t}$ を下記のように定義する．

$$c_{h,k,t} = f({}^T\mathbf{w}_{h,k}\mathbf{x}^{t:t+h-1} + b_{h,k}) \quad (5)$$

ここで， $b_{h,k} \in \mathbb{R}$ は h 個の形態素における k 番目のフィルタのバイアス項を表す． ${}^T\mathbf{w}_{h,k}$ は $\mathbf{w}_{h,k}$ の横ベクトルを表す．また， f は非線形関数で ReLU を利用する．

次に 1 文が N 個の形態素からなるとき，上記の特徴量に対して 1 文で最大値のみをとるプーリング (max-pooling) を適用する [3]．まず N 個の形態素に対して上記の式 (5) を適用して得られる値を $c_{h,k}$ として下記のように定義する．

$$c_{h,k} = {}^T[c_{h,k,1}, \dots, c_{h,k,t}, \dots, c_{h,k,N-h+1}] \quad (6)$$

式 (5) の $c_{h,k}$ に対して下記のように max-pooling を適用し，1 文に対する特徴量 $c_{h,k}$ を得る．

$$c_{h,k} = \max_{t=1, \dots, N-h+1} \{c_{h,k,t}\} \quad (7)$$

次に各 $c_{h,k}$ に対してすべての h と k を組み合わせて得られる値を結合したベクトル $\mathbf{c} \in \mathbb{R}^{HK}$ を仮定する．

$$\mathbf{c} = c_{h_{min},1} \oplus \dots \oplus c_{h_{min},K} \oplus \dots \oplus c_{h_{max},K} \quad (8)$$

ここで $H = h_{max} - h_{min} + 1$ とする*10．式 (8) の $\mathbf{c}$ に対し

*9 たとえば $h = 3$ 個の形態素で $d = 200$ 次元ならば $\mathbf{w}_{h,k}$ は 600 次元のベクトルを表す．

*10 たとえば 3 形態素から 5 形態素まで h を変化させた場合， $H = 3$ となり， $K = 128$ 個のフィルタを用意した場合 $\mathbf{c}$ は $3 \times 128 = 384$ 次元のベクトルを表す．

*7 本論文では上付きの左に付く T で転置行列を表す．

*8 入力順序について 4.3 節で説明する．

て式 (9) を適用することで出力ベクトル $\mathbf{y} \in \mathbb{R}^{Sem}$ を得る.

$$\mathbf{y} = f_{out}(\mathbf{w}_{out}\mathbf{c} + \mathbf{b}_{out}) \quad (9)$$

また $\mathbf{w}_{out} \in \mathbb{R}^{Sem \times HK}$ は最終出力層への結合重みを表し, $\mathbf{b}_{out} \in \mathbb{R}^{Sem}$ はそのバイアス項を表す. 非線形関数 f_{out} は softmax 関数を適用する. 誤差関数には交差エントロピー誤差を適用し, 最適化では Adam を用いる.

3.3 GRU モデル

GRU を利用した意味役割付与モデルは時刻 t における内部状態を $\mathbf{h}^t \in \mathbb{R}^m$ とすると最終出力 $\mathbf{y} \in \mathbb{R}^{Sem}$ を最終時刻 T の状態 $\mathbf{h}^T$ を利用して下記のように求める.

$$\mathbf{y} = f_{out}(\mathbf{W}_{out}\mathbf{h}^T + \mathbf{b}_{out}) \quad (10)$$

ここで, $\mathbf{W}_{out} \in \mathbb{R}^{Sem \times m}$ は最終出力層への結合重みを表し, $\mathbf{b}_{out} \in \mathbb{R}^{Sem}$ はバイアス項を表す. f_{out} は softmax 関数を利用する.

時系列として $\mathbf{x}^1, \dots, \mathbf{x}^t, \dots, \mathbf{x}^T$ ($\mathbf{x}^t \in \mathbb{R}^d$) が入力されたとき, 内部状態 $\mathbf{h}^t$ は下記のように求める [36].

$$\mathbf{h}^t = (1 - \mathbf{z}^t) \odot \mathbf{h}^{t-1} + \mathbf{z}^t \odot \tilde{\mathbf{h}}^t \quad (11)$$

$$\tilde{\mathbf{h}}^t = \tanh(\mathbf{W}_h \mathbf{x}^t + \mathbf{b}_h + \mathbf{W}_{hh}(\mathbf{r}^t \odot \mathbf{h}^{t-1}) + \mathbf{b}_{hh}) \quad (12)$$

$$\mathbf{z}^t = \sigma(\mathbf{W}_z \mathbf{x}^t + \mathbf{b}_z + \mathbf{W}_{zh} \mathbf{h}^{t-1} + \mathbf{b}_{zh}) \quad (13)$$

$$\mathbf{r}^t = \sigma(\mathbf{W}_r \mathbf{x}^t + \mathbf{b}_r + \mathbf{W}_{rh} \mathbf{h}^{t-1} + \mathbf{b}_{rh}) \quad (14)$$

ここで $\tilde{\mathbf{h}} \in \mathbb{R}^m$ は出力候補, $\mathbf{z}^t \in \mathbb{R}^m$, $\mathbf{r}^t \in \mathbb{R}^m$ はアップデートゲート, リセットゲートを表しており, $\odot$ はベクトルどうしの要素積を表している. 各 $\mathbf{W}_h$, $\mathbf{W}_z$, $\mathbf{W}_r$ は $\mathbb{R}^{m \times d}$ の重みであり, $\mathbf{W}_{hh}$, $\mathbf{W}_{zh}$, $\mathbf{W}_{rh}$ は $\mathbb{R}^{m \times m}$ の重みである. 各 $\mathbf{b}_h$, $\mathbf{b}_{hh}$, $\mathbf{b}_z$, $\mathbf{b}_r$, $\mathbf{b}_{zh}$, $\mathbf{b}_{rh}$ は $\mathbb{R}^m$ のバイアス項である. また活性化関数として tanh は双曲線関数を表し, σ はシグモイド関数を表す.

誤差関数には交差エントロピー, 最適化では Adam を利用する. 入力として意味役割を推定する項と述語の形態素に関する分散表現ベクトルを時系列に並べたものを仮定する. 時系列の並べ方やハイパーパラメータなどの詳細は 4.3 節に記述する.

3.4 双方向 GRU max-pooling モデル (Bi-GRU)

文献 [4] において複数の分類タスクで高い精度を示した Bi-LSTM max-pooling モデルを参考に GRU に置き換えたモデルを利用する. 図 3 に示すように, 前方向と後方向に接続した GRU をそれぞれ利用する. 3.3 節で定義した GRU を前方向 GRU とし, その内部状態を $\vec{\mathbf{h}}^t$ で表す. 一方で後方向に接続した GRU は式 (12) から式 (14) の右辺の $\mathbf{h}^{t-1}$ を $\mathbf{h}^{t+1}$ に置き換えることで定式化できる. このときの GRU の内部状態を $\overleftarrow{\mathbf{h}}^t$ で表す.

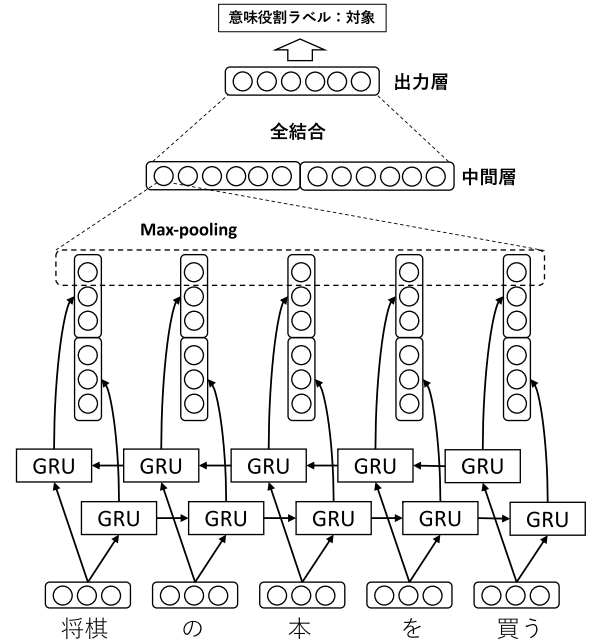


図 3 Bi-GRU モデル

Fig. 3 Bi-GRU model.

前方向と後方向それぞれの内部状態を結合したベクトル $\mathbf{h}^t \in \mathbb{R}^{2m}$ を作成する.

$$\mathbf{h}^t = \vec{\mathbf{h}}^t \oplus \overleftarrow{\mathbf{h}}^t \quad (15)$$

各 t 番目の形態素で作成した $\mathbf{h}^t$ に対して, 今度は時系列方向を横断する形で, j 番目の内部状態を取り出したベクトル $\mathbf{m}_j \in \mathbb{R}^T$ を考える.

$$\mathbf{m}_j = [h_j^1, \dots, h_j^t, \dots, h_j^T] \quad (16)$$

$\mathbf{m}_j$ の時系列要素に対して max-pooling を適用した値 m_j を下記のように求める.

$$m_j = \max_{t=1, \dots, T} \{\mathbf{m}_j\} \quad (17)$$

これをすべての j について適用して得られたベクトル $\mathbf{m} \in \mathbb{R}^{2m}$ を下記のように定義する.

$$\mathbf{m} = [m_1, \dots, m_{2m}] \quad (18)$$

得られた $\mathbf{m}$ に対して $\mathbf{W}_{out} \in \mathbb{R}^{Sem \times 2m}$ を適用することで最終出力 $\mathbf{y}$ を得る.

$$\mathbf{y} = f_{out}(\mathbf{W}_{out}\mathbf{m} + \mathbf{b}_{out}) \quad (19)$$

ここで $\mathbf{b}_{out}$ は $\mathbb{R}^{Sem}$ のバイアス項を表す. また f_{out} は softmax 関数を適用する. 誤差関数は交差エントロピー誤差を適用し, 最適化には Adam を用いる.

3.5 転移学習の利用

本論文で利用する転移学習 [32] とは, 学習対象とする分類クラスと異なるクラスを学習したニューラルネットワークを利用して, 最終出力層のみ取り替えて, 対象とする分

類クラスを学習させることで、まったくの乱数で初期化した重みを利用するよりも精度の向上を期待する手法である。これは分類クラスは異なっているが、下位層にタスクに共通する特徴量が学習で獲得できる場合に有効となる。

学習対象とする BCCWJ-PT に対して、意味役割ラベルは異なるが関連するラベルを付与している新聞記事 GDA コーパス 2004 [35]^{*11}を転移学習における最初の学習データとして利用する。GDA に付与されている関係子では意味役割に対応する「動作主 (agt)」や「対象 (obj)」などが定義されている一方で、BCCWJ-PT にはない主題化 (たとえば「topic」) や文法情報 (たとえば「sbj」) など詳細な分類ラベル (約 100 種類) が付与されている。また学習データが BCCWJ-PT に比べて多いため、ニューラルネットワーク内に意味役割付与に必要な特徴量が学習されることが期待できる。

転移学習は上記で説明したニューラルネットワークを利用した意味役割付与モデルすべてに適用する。手順としてはまず、最終出力層の結合重み式 (2), (9), (10), (19) の W_{out} と b_{out} を GDA の関係子の数に合わせて GDA コーパスで全体の重みを学習する。次に、 W_{out} と b_{out} のみ初期化して BCCWJ-PT の意味役割ラベルの数に合わせて BCCWJ-PT で学習する。これにより精度の向上を目指す。

4. 意味役割付与実験

BCCWJ-PT データに対して提案した意味役割モデルを適用して付与実験を行う。下記では意味役割付与データ、転移学習に利用する GDA コーパスを利用したデータ、実験設定、評価方法について説明し、実験結果について述べる。

4.1 意味役割付与データ

2 節で記述したように BCCWJ-PT コーパスを項ごとに意味役割を付与したデータに変換して意味役割実験を行う。よって転移学習で利用する GDA コーパスも同様に対象となる述語に対して項と意味役割ラベルを取り出して項単位の事例データを作成する。GDA コーパスでは関係子は「topic.fit.ctl.gol」など組み合わせで項の意味的な関係を表している場合がある。本研究では組合せを 1 つの意味的な関係と見なして、395 種類を意味的な関係として扱う。

抽出後のデータに対して UniDic 辞書^{*12}の形態素解析器 MeCab^{*13}を利用して表層形で形態素ごとに分割する。また述語部分は基本形も取り出す。こうして抽出したデータを以降 BCCWJ-PT データ、GDA データと呼ぶ。

表 2 に BCCWJ-PT データ、および GDA データの学習

表 2 実験に利用するデータ量

Table 2 Amount of the data used in the experiments.

コーパス	全体	学習	開発	テスト
BCCWJ-PT	10,390	6,753	520	3,117
GDA	82,892	70,458	4,144	8,290

表 3 使用するコーパスに含まれる意味役割ラベル上位 10 件

Table 3 The top 10 semantic role labels in the corpus.

意味役割	事例数
対象	3,101
動作主	1,201
時間	586
様態	549
修飾	451
副詞	441
経験者	418
着点	407
場所	327
原因	309

データ、開発データ、テストデータの量を示す。BCCWJ-PT データでは学習データ、開発データ、テストデータの比率は 65%, 5%, 30%であるのに対して、転移学習で利用する GDA では 85%, 5%, 10%と学習データを大きめに設定している。これは GDA データでは特徴量を獲得することが目標であるため多くの学習データを利用することを想定したためである。

BCCWJ-PT データで定義されている意味役割ラベルの詳細な説明と事例は述語シソーラスサイトに記述されている^{*14}。本研究で利用した BCCWJ-PT0.91 に含まれている意味役割は 62 種類ある。そのうち上位 10 件を表 3 に示す。特に「対象」と「動作主」が多く、意味役割ごとに事例の頻度が大きく異なるのが特徴である。

4.2 ベースラインモデル

ベースラインモデルとして SVM を利用する。多項式カーネルを利用し^{*15}、one-vs-rest 法により 62 種類の意味役割ラベルを識別する。パッケージとして scikit-learn を利用した。SVM のハイパーパラメータ C について Python のパッケージである GPyOpt を利用し、開発データを利用して最適化した^{*16}。入力として 4.3 節で記述するように 3LNN と同じ特徴量を利用した。

4.3 入力で利用する特徴量

意味役割付与モデルに入力の異なりによる付与精度への影響を調べるために異なる特徴量の組合せを定義する。

^{*11} 関係子に関する説明は <http://i-content.org/gda/tagman.html>。また GDA コーパスの配付は <https://www.gsk.or.jp/catalog/gsk2009-b>。

^{*12} <https://unidic.ninjal.ac.jp/>。

^{*13} <http://taku910.github.io/mecab/>。

^{*14} <http://pth.cl.cs.okayama-u.ac.jp/>。

^{*15} 線形、多項式、RBF カーネルについて予備実験を行い、多項式カーネルが最も有効であった。

^{*16} 最適化の結果、 $C = 1.0$, $\gamma = 0.1$ とした。

表 4 SVM および 3LNN で利用する基本特徴量

Table 4 The base features applied to SVM and 3LNN.

特徴量	次元数	説明
bow	10,422	項の形態素と述語の基本形と表層の bag-of-words
phsk	400	項の主辞と述語の基本形の分散表現
two	5,206	項の末尾 2 形態素の bag-of-words
dv	200	項の形態素と述語の基本形と表層の分散表現の平均

時系列を扱わない SVM と 3LNN に対して時系列を扱う CNN, GRU, Bi-GRU では入力ベクトルの処理が異なるため, 特徴量を分けて定義する.

SVM と 3LNN について組み合わせて利用する基本特徴量と各次元を表 4 に示す. 表 4 の bow は, たとえば表 1 の 2 行目の事例に対して, 「太郎」, 「は」, 「買う」, 「買う」の 4 つの形態素に対応した座標のみ 1 で他は 0 の 10,422 次元のベクトルとなる. phsk は国立国語研究所が日本語ウェブコーパスから作成した分散表現 nwjc2vec [37] を利用して, 項の主辞の形態素, および, 述語の基本形の skip-gram ベクトル^{*17} (200 次元) を結合したベクトルを表す. また, two は先行研究 [38] で有効性が示されている項の末尾 2 形態素^{*18}に対して, 2 形態素の bag-of-words によるベクトルを表す. dv も nwjc2vec の skip-gram を利用する. dv は, 具体的には表 1 の 2 行目の事例では「太郎」, 「は」, 「買う」, 「買う」の 4 つの形態素の skip-gram ベクトルを加えて平均したベクトルである.

時系列モデルでは, すべての形態素を分散表現に変換して入力する. 分散表現ベクトルは上記と同様に nwjc2vec の skip-gram (200 次元) を利用する. 入力順の違いによる精度への影響が予測されるため, CNN, GRU, BiGRU モデルでは表 5 に示すように順序の異なる特徴量を仮定した. GRU モデルでは表 5 の v1 から v4 までのすべての特徴量に対して評価実験を行った結果を 4.6 節で記述する. 一方で, CNN では予備実験で最も精度が高かった特徴量 v1 を利用し, BiGRU モデルでは GRU で最も精度が高かった v2 を利用する.

4.4 実験設定

SVM を含むすべてのモデルは学習データで学習を行い, テストデータで評価する. SVM では開発データをハイパーパラメータの調整に利用した. ニューラルネットワークモデルでは, 開発データを利用して, 学習回数 (epoch) を決めた. また, ニューラルネットワークの実装は Tensor-

表 5 CNN, GRU, BiGRU モデルで利用する特徴量

Table 5 Features applied to CNN, GRU and BiGRU.

特徴量	説明	例
v1	項の後に動詞の基本形	詰め将棋 / の / 本 / を / 買う
v2	項の前に動詞の基本形	買う / 詰め将棋 / の / 本 / を
v3	項の前に動詞の表層の表層	買う / 詰め将棋 / の / 本 / を
v4	項の前に動詞の表層と基本形	買う / 買う / 詰め将棋 / の / 本 / を

Flow^{*19}を利用した. 各モデルは 1 事例に対して, 推定した 1 つの意味役割ラベルのみ出力する. 評価はテストデータに対する正解率で評価する.

$$\text{正解率} = \frac{\text{推定した意味役割が正解と一致した事例数}}{\text{全事例数}} \quad (20)$$

4.5 実験における各モデルのパラメータ

ニューラルネットワークの中間層は 256 次元とする. また, 最適化で利用する Adam のパラメータは $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$ とする. ドロップアウトは 0.5 を設定し, 3LNN モデルのみ適用する. epoch は開発データによる観測から 20 とし, 転移学習時の GDA データに対する epoch および BCCWJ-PT に対する epoch も 20 とする.

畳み込みに用いるフィルタの窓幅として連続する 3, 4, 5 形態素の 3 種類を設定し, それぞれフィルタを 128 枚適用する.

4.6 BCCWJ-PT データに対する意味役割付与実験の結果と評価

表 6 に BCCWJ-PT データに対して学習およびテストした意味役割付与精度の結果を示す. まず, モデルの異なることによる正解率を比較すると, BiGRU モデルがすべてのモデルの中で最も高い正解率を示した. BiGRU と各モデルとの McNemar 検定を行ったところ, SVM に対して $p = 6.00 \times 10^{-3}$, 3LNN に対して $p = 1.69 \times 10^{-7}$, GRU に対して $p = 6.15 \times 10^{-13}$, CNN に対して $p = 1.02 \times 10^{-3}$ となりすべてのモデルの結果に対して 5%水準で有意であった.

次に特徴量と各モデルの出力に関して比較する. ベースラインである SVM と 3LNN の特徴量について比較すると, 項の主辞と述語の skip-gram (phsk) の適用で各々約 10%と約 6%, 末尾形態素 (two) の適用で約 4%と約 3%, 分散表現の平均ベクトル (dv) の適用で約 2%と約 1%, 正

^{*17} https://pj.ninjal.ac.jp/corpus_center/nwjc/subscription.html で配付されている.

^{*18} 項の末尾に現れる「に対して」「なので」など複合辞や助詞のによる表現を考慮して, 末尾の 2 形態素を特徴量として利用すると意味役割で精度が向上することが示されている [38].

^{*19} <https://www.tensorflow.org/>.

表 6 各モデルにおける異なる特徴量に対する識別精度

Table 6 Accuracies of the models with different features.

モデル	特徴量	正解率
SVM	BOW	0.494
	BOW + phsk	0.604
	BOW + phsk + two	0.642
	BOW + phsk + two + dv	0.664
3LNN	BOW	0.551
	BOW + phsk	0.622
	BOW + phsk + two	0.653
	BOW + phsk + two + dv	0.661
CNN	v1	0.678
	v1 + BOW	0.687
	v1 + BOW + phsk	0.680
	v1 + BOW + phsk + two	0.679
GRU	v1	0.615
	v2	0.632
	v3	0.592
	v4	0.626
	v2 + BOW	0.648
	v2 + BOW + phsk	0.640
	v2 + BOW + phsk + two	0.648
BiGRU	v2	0.702*
	v2 + BOW	0.693
	v2 + BOW + phsk	0.683
	v2 + BOW + phsk + two	0.684

解率が向上し有効に機能していることが分かる。

SVM と 3LNN を比較すると同じ特徴量を利用した場合には分散表現の平均ベクトル (dv) を利用しない場合は 3LNN の方が高い正解率を示したが、dv を利用した場合は SVM の方が高い正解率を示した。また、GRU の正解率を超える精度を示している。このことから特徴量の組合せによって、SVM が 3LNN や GRU を越える精度を示すことが分かる。

次に時系列モデルの特徴量について比較する。入力の手順による正解率への影響を GRU で調べたところ v2 の特徴量が最も高かった。このことから動詞の基本形を利用することと項が最終に来る順序が有効であることが分かる。予備実験により CNN は v1、BiGRU の場合は v2 が正解率が高かったため、この 2 つのモデルでは以降、v1 および v2 を利用する。

CNN の場合は形態素の連続列に対する特徴量を取り出すため、v1 の場合「本を買う」といった意味役割に直接影響を与える項の末尾表現と述語の部分が連続して取り出されていることが要因の 1 つとして考えられる。一方で、v2 の場合 GRU と BiGRU では項の末尾表現が時系列の最後に来る (BiGRU の場合は forward のとき) ことにより意味役割の決定に有効に働いたと考えられる。

特徴量 BOW を追加した場合、CNN、GRU、BiGRU モデルでは、特徴量の追加が必ずしも精度の向上に結び付い

表 7 各モデルにおける転移学習による識別精度

Table 7 Accuracies of the transfer learned model.

モデル	特徴量	正解率	
		転移なし	転移あり
3LNN	BOW + phsk + two + dv	0.661	0.683*
CNN	v1 + BOW	0.687	0.680
GRU	v2 + BOW + phsk + two	0.648	0.660*
BiGRU	v2	0.702	0.687

ていない。BOW の追加では CNN、GRU モデルとも正解率の向上が見られたが、BiGRU モデルでは正解率が低下した。これは CNN と GRU では BOW による順序に依存しない単語全体の特徴量が有効に機能したと考えられる。一方で、BiGRU では時系列を横断して max-pooling を行う層があることから、順序によらない全体的な単語の出現に関する特徴量がモデルの中で生成されていることが考えられる。

次に phsk の追加では CNN、GRU、BiGRU モデルのいずれにおいても BOW を加えた場合に比べて正解率を下げる結果を得た。これらのモデルでは入力ですでに skip-gram を利用しているため、項の主辞と述語について強調した特徴量を phsk で追加した形になっているが、連続列の中で構築される特徴量の方が部分的に形態素を取り出して加えるよりも有効に機能していると考えられる。

また two の追加では CNN と BiGRU では BOW+phsk に対して正解率がほとんど変わらず、GRU では 0.8% 上昇したものの、正解率は BOW のみを加えた場合と同程度となった。このことから、末尾表現の特徴量の追加は時系列モデルでは有効に働かないことが分かる。これは上記で述べたように、CNN では形態素連続列として v1 の場合に末尾特徴量が考慮されていること、GRU および BiGRU モデルでは v2 に末尾情報が取り込めていることから末尾情報を追加しても正解率の向上が得られなかったのではないかと考えられる。

4.7 GDA データによる転移学習を利用した実験結果と評価

各ニューラルネットワークのモデルに対して、転移学習を適用した意味役割付与実験の結果を表 7 に示す。各モデルでは表 6 で最も高い正解率を示した特徴量を利用した。表 6 と比較して、転移学習後の正解率は 3LNN、GRU モデルでは正解率は向上したものの、CNN、BiGRU では逆に正解率の低下が見られた。正解率が向上した 3LNN および GRU の結果に対して転移学習の有無で McNemar 検定で評価したところ p 値はそれぞれ $p = 7.9 \times 10^{-5}$ 、 $p = 0.039$ となり ($p < 0.05$) で 5% 水準で有意であった (表 7 に*印で表記)。このことから、これらの 2 つのモデルでは転移学習が正解率の向上に有効に働いていることが分かる。

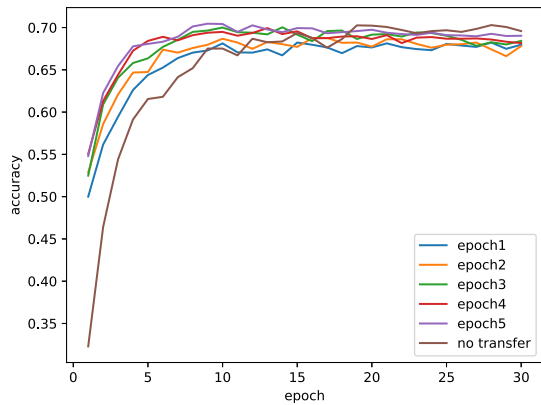


図 4 転移学習における初期学習データ (GDA データ) の epoch1 から 5 の場合の BCCWJ-PT に対する epoch と正解率の変化
Fig. 4 Comparing the models in their accuracies for learning BCCWJ-PT changing the initial learning epochs from 1 to 5 for GDA.

一方で, 3LNN, GRU の転移学習により向上した正解率よりも, BCCWJ-PT データのみを学習した場合の CNN と BiGRU の正解率が高い. CNN と BiGRU は表 6 に示すように元々 3LNN と GRU よりも精度が高く, BCCWJ-PT データのみで特徴を取り出せていると考えられる. よってネットワークの構造によっては目的データと異なる正解ラベルに対する特徴量を過度に学習してしまい目的データへの適合がうまくいかなかったと考えられる.

よって上記の実験設定で最も高い正解率を示したモデルは転移学習を用いない BiGRU である. しかしながら, 特徴量を過度に学習する可能性があるならば, GDA データによる epoch を短くして過学習を防いだ場合には異なる結果が得られる可能性がある. また, 最終の正解率だけではなく, GDA データを学習させた場合と初期値を乱数で設定した場合では収束に対して異なる影響が考えられる. そこで次節で GDA データに対する epoch を少なくした場合の実験を行い結果について考察する.

4.8 転移学習における GDA データに対する epoch の異なりによる収束の考察

4.7 節における GDA データの学習では epoch を 20 で固定した. 本節では epoch を 1 から 10 に変化させた場合の転移学習の結果について評価する. GDA で学習させた後, 転移学習で BCCWJ-PT の学習データで学習し, テストデータで評価する. BCCWJ-PT の epoch を横軸とし, BCCWJ-PT のテストデータに対する正解率 (accuracy) を縦軸とした評価結果を図 4 と図 5 に示す. 図 4 は GDA データの epoch が 1 から 5 までの場合, 図 5 は epoch が 6 から 10 の場合である.

ここで図 4, 図 5 内の「no transfer」は転移学習なしの場合を示す. まず横軸に対する正解率の収束状況を見ると, 転移学習しない場合に比べて, 転移学習を適用した場合は

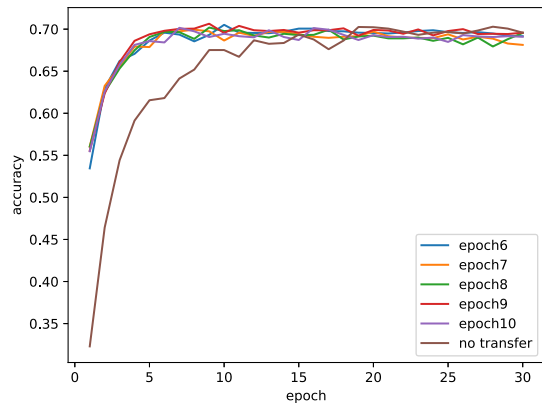


図 5 転移学習における初期学習データ (GDA データ) の epoch6 から 10 の場合の BCCWJ-PT に対する epoch と正解率の変化
Fig. 5 Comparing the models in their accuracies for learning BCCWJ-PT changing the initial learning epochs from 6 to 10 for GDA.

表 8 転移学習による GDA データに対する epoch と BCCWJ-PT データでの最高正解率の epoch

Table 8 Epochs for the best accuracy in learning BCCWJ-PT with the epochs in the initial GDA learning.

GDA 学習時の epoch	正解率 (BCCWJ-PT 学習時の epoch)
転移学習なし	0.703 (19)
1	0.682 (15)
2	0.688 (17)
3	0.700 (10)
4	0.699 (13)
5	0.705 (9)
6	0.705 (10)
7	0.699 (8)
8	0.702 (9)
9	0.706 (9)
10	0.702 (7)

BCCWJ-PT に対して少ない epoch で正解率が上昇することが分かる. 図 4 において, GDA に対する epoch で比較すると, GDA の epoch が 1 から 5 までは epoch の増加に応じて正解率が早く上昇することが分かる. 一方, 図 5 では GDA の epoch が 9 (赤) の場合が比較的, 正解率が高く, epoch が 10 (紫) から下回る傾向にある. このことから, BiGRU モデルにおいても, 転移学習による GDA の学習から, 一定の有益な特徴量を初期値として獲得しており, 目的データである BCCWJ-PT の学習で, 早く正解率を上昇させる効果があることが分かる.

一方で, GDA データで epoch が多いと最終的な正解率は低下した (表 7). そこで, もし適切に学習回数を打ち切ることができた場合の転移学習の効果について調べる. 表 8 に GDA データの epoch が 1 から 10 の場合に, BCCWJ-PT のテストデータに対して最も正解率が高かつ

表 9 各意味役割ラベルに対する識別精度

Table 9 Accuracies of the models for each semantic role label.

ラベル	SVM	3LNN (転移)	GRU (転移)	CNN	BiGRU
対象	0.890	0.886	0.846	0.889	0.877
動作主	0.782	0.800	0.785	0.758	0.797
時間	0.731	0.791	0.806	0.853	0.853
様態	0.561	0.602	0.593	0.585	0.724
修飾	0.522	0.507	0.574	0.610	0.669
副詞	0.382	0.317	0.327	0.366	0.347
経験者	0.565	0.679	0.603	0.649	0.710
着点	0.618	0.636	0.564	0.627	0.618
場所	0.652	0.750	0.682	0.727	0.697
原因	0.277	0.328	0.422	0.406	0.438

表 10 テストデータ内の意味役割ラベルに対する末尾表現の内訳

Table 10 Details of case markers or final expressions for semantic role labels.

ラベル	格助詞			係助詞 は, も, 他	その他	合計 数
	が	を	他			
対象	0.154	0.51	0.043	0.103	0.190	928
動作主	0.264	0	0.053	0.322	0.361	413

た場合の BCCWJ-PT の epoch について示す*20. 表 8 から正解率は転移学習がなかった場合と比べて GDA の epoch が 5, 6, 9 のときにわずかながら高い値を示す場合があることが分かる. また, 傾向として GDA の epoch が増加するにつれて, BCCWJ-PT の学習時の epoch は少なくなることが分かる. 表 8 の結果は BCCWJ-PT の学習時にテストデータの正解率で学習回数を決定しているため, 実際の状況では得られない理想的な正解率である. しかしながら目的データである BCCWJ-PT の意味役割ラベルと部分的に対応する GDA の関係子が転移学習により有効に機能する可能性があることを示唆していると考えられる.

4.9 意味役割ラベルの精度評価

表 3 に示した上位 10 件の意味役割ラベルを対象に各モデルの識別精度を表 9 に示す. 各モデルの特徴量は表 6 で最も高い正解率を示した特徴量を利用し, 3LNN と GRU については転移学習を適用したモデルを利用する.

まず, 出現頻度に対する正解率を比較すると頻度が最も高い「対象」はどのモデルでも高い正解率を示した. しかしながら「動作主」以下では必ずしも出現頻度が正解率と比例関係にはない. たとえば「時間」は「動作主」の半数以下の出現頻度であるが, GRU, CNN, BiGRU では「動作主」より正解率が高い. よって頻度よりも意味役割の特徴をいかにうまく取り出せるかに依存する.

ここで表 10 にテストデータ内「対象」と「動作主」に

*20 BCCWJ-PT の epoch で 20 までの範囲とする.

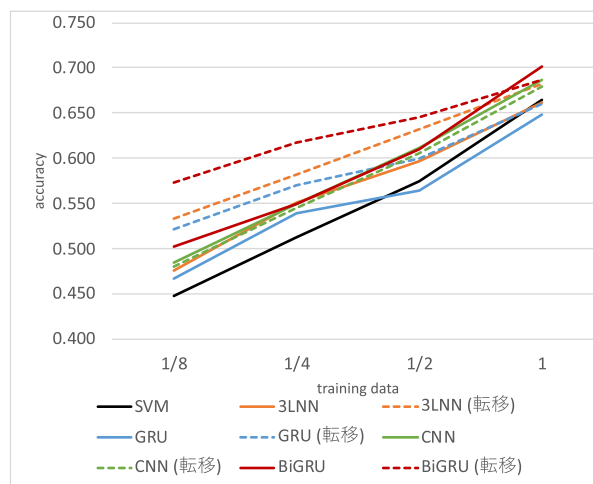


図 6 学習データ量に対する各モデルの識別精度

Fig. 6 Accuracies of the models changing the amount of the learning data.

おける項の末尾表現の内訳を示す*21. 対象の場合, 約 5 割が「を」であるが, 他は「動作主」にも存在する「が」や係助詞, 他の品詞が存在するため, 助詞などの単純な手がかりだけでは, 識別することが容易ではないことが分かる.

表 9 でモデルどうしの正解率を比較すると, GRU に対してはすべての意味役割で BiGRU の方がすべての意味役割で高い正解率を示した. 一方で, SVM, 3LNN, CNN に対しては BiGRU の方が高い正解率を示す場合が多いが, 一部の意味役割では SVM, 3LNN や CNN が高い正解率を示した. このことから, GRU を取り込んだ BiGRU は GRU の特徴は取り込んでいる一方で, SVM, 3LNN や CNN で識別できている特徴がうまく取り込めていないことから, まだ改善の余地があることが分かる.

4.10 学習データ量に関する各モデルの精度

学習データ量に対してどの程度各モデルが特徴を獲得できるかを明らかにするために, 学習データ量を変化させた場合の精度を比較する.

図 6 は元の学習データ (training data) を 1 とし, 1/2, 1/4, 1/8 と減少させた場合の各モデルにおけるテストデータに対する正解率 (accuracy) を示している. 各モデルの特徴量は表 6 で最も高い正解率を示した特徴量を利用した. SVM 以外のモデルでは 4.7 節で実行した転移学習を適用したモデルについても実験を行い, 評価結果を同じ色の破線で示している.

SVM とニューラルネットワークモデルを比較すると, GRU を除いて, 学習データが減少した場合にニューラルネットワークモデルの正解率が一貫して高いことが分かる. GRU モデルも学習データが 1/4, 1/8 と小さくなった場合には SVM を上回る傾向が観測された. 最も精度が高

*21 表内の数値は合計数に対する割合を表す.

かった BiGRU は SVM と比較してもほぼ同様の傾向で正解率が減少するため、SVM に対する精度の優位性は学習データが減少しても変わらないことが分かる。こうした結果から、提案するニューラルネットワークモデルは SVM に対して効率的な学習が行えていることが考えられる。

次に、転移学習の効果について比較すると、CNN を除いた、3LNN, GRU, BiGRU に対して転移学習を適用したモデルの方が学習データの減少に対して識別精度の低下が少ない傾向が見られた。よって、モデルに依存するが、転移学習が学習データが少ない場合に識別精度の向上に寄与する可能性が高いことが考えられる。

全学習データで最も高い精度を示した BiGRU は学習データが 1/2 に減少した段階で BiGRU (転移) の方が精度が高くなり、学習データが減少するに従い、BiGRU の識別精度との差が開いていく傾向が見られた。学習データが 1/2 以下の場合にはすべてのモデルの中で BiGRU (転移) が最も高い識別精度を示していることから、BiGRU のネットワーク構造で、転移学習を適用した場合には、効率的な学習が行えることが考えられる。

5. 関連研究

意味役割付与研究の初期段階では主に統計モデルを利用して効果的な特徴量が研究されてきた。文献 [8] では FrameNet に対して構文解析を利用した文法的な特徴量を利用して各項に対する意味役割のラベル付与を行った。文献 [23] では PropBank を構築し、文献 [8] と同様の手法を適用した。しかしながら、構文解析器の誤りの影響が指摘され [25], [30]、近年のニューラルネットワークを利用して構文解析を利用しない End-to-End の手法が Collobert らの研究 [3] を初めとして提案されるようになった。

ニューラルネットワークの構造として複数の双方向 RNN を利用したモデルが多く利用され高い識別結果を示している [10], [11], [22], [24], [31]。文献 [31] では双方向 LSTM に CRF を適用して系列ラベル問題として意味役割付与タスクを実行した。文献 [11] では双方向 LSTM に highway 構造を取り入れてアンサンブル学習を利用し文献 [31] の精度を上回った。Peters ら [24] は He ら [11] のモデルに新たな分散表現である Elmo を利用することでさらに精度を向上することを明らかにした。Ouchi ら [22] や He ら [10] は意味役割を付与する範囲を推定するスパンモデル導入することで意味役割付与精度を向上させることを示した。また RNN を利用しないモデルとして Tan ら [29] は self-attention を利用したモデルを提案し He ら [11] のモデルよりも精度が向上する場合があることを示した。さらに Strubell ら [28] は self-attention に Dozat ら [6] の係り受け解析モデルを取り込むことで He ら [10] の精度を超える結果を示している。

これらのタスクでは項の範囲と意味役割ラベルを同時に推定しており、本論文のタスクでは項の範囲を既知として

与えているため、上記の提案モデルとタスクが異なるが、本論文の実験結果でも双方向 RNN を利用したモデルは意味役割付与に対して高い精度を示しており、上記の研究結果と同様の傾向にある。一方で、本論文では扱っていない新たなニューラルネットワークの構造として self-attention, highway, スパンモデル、さらに分散表現として Elmo などの利用が提案されておりこれらの利用は今後の課題としたい。

日本語の述語項構造解析では、項と述語のグラフ構造を利用した手法 [20], [27] や、ニューラルネットワークに複数の述語との関係を取り込むことで、識別精度を向上させる手法が提案されている [18], [21]。松林ら [18] は双方向 GRU モデルと max-pooling および attention 機構を導入したモデルを提案し識別精度が向上することを示している。本研究で利用しているモデルは attention 機構を除いて同様の構造であることから、双方向 RNN と max-pooling の組合せは述語項構造解析、および意味役割ラベルの決定に有効に働くことが分かる。本研究の意味役割付与タスクにおける attention 機構の取り入れ方などは今後の課題としたい。

意味役割付与タスクにおいて正解データが少ない場合の先行研究として既存の意味役割付与データを利用して学習データを拡張する手法が提案されている [5], [7]。しかしながら、これらの手法では対象とする意味役割ラベルセットが同一である。本論文で提案した転移学習は付与対象のラベルセットが異なる学習データを利用して意味役割付与の精度向上を行う手法であるため、提案手法の方が先行研究に比べて適用範囲が広いと考えられる。

BCCWJ-PT を対象とした意味役割付与の先行研究として末尾表現の特徴量を利用した石原ら [38] の手法があるが 10 分割交差検定で F 値が 60.53% であった。一方で本提案手法では F 値で 70.2% と大きく上回っている*22。さらに本論文の実験設定では学習データが 65% でテストデータが 30% であり、先行研究に比べて学習データの比率が小さい。このことからデータセットが異なるため直接比較はできないが、本提案手法が先行研究に比べて有効であると考えられる。

6. おわりに

本論文では日本語意味役割ラベル付与タスクに対して、ニューラルネットワークを適用し、従来の特徴量を利用した手法 [38] を上回る精度を示すことを実験的に明らかにした。ニューラルネットワークとして 3 層ニューラルネットワーク、畳込みニューラルネットワーク、GRU、双方向 GRU を適用した結果、双方向 GRU モデルが最も高い識別精度を示すことを明らかにした。また入力における特徴量

*22 比較のために正解率を適合率および再現率として F 値を求めた。

の影響として、先行研究で示された文法的特徴量は識別精度の向上に有効に寄与しないこと、ならびに、時系列を扱うニューラルネットワークでは、述語と項の順で並べた場合が有効に働くことを示した。

さらに、目的とする意味役割体系と異なる他の意味役割が付与されたデータを利用して転移学習を適用し、識別精度を向上を試みた。その結果、ニューラルネットワークの構造によって識別精度が向上するものと下降するものがあることを明らかにした。また、転移学習前で最も精度が高い双方向 GRU モデルの場合に転移学習を適用した場合に、識別精度の大きな向上は見られなかったが、目的データでの収束回数が短くなること、ならびに、適切な学習回数が分かれば、さらに精度向上が見込まれることを明らかにした。転移学習の効果として、CNN 以外のニューラルネットワークモデルで学習データが減少した場合に識別精度の低下を和らげる傾向があることが実験的に明らかになった。

今後の課題として、英語の意味役割付与タスクと同様に項の範囲も同時に推定する日本語意味役割付与タスクを設定し、有効なモデル化について明らかにしたい。

謝辞 本研究の一部は国立大学法人運営費交付金（機能強化経費）および JSPS 科研費 19K00552 の助成を受けた。査読者の皆様、ならびにメタ査読者から大変有益なコメントをいただいたことに感謝する。

参考文献

- [1] Baker, C.F., Fillmore, C.J. and Lowe, J.B.: The Berkeley FrameNet project, *Proc. 36th Annual Meeting of the Association for Computational Linguistics*, pp.86–90 (1998).
- [2] Carreras, X. and Màrquez, L.: Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling, *Proc. CoNLL-2005 Shared Task* (2005).
- [3] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K. and Kuksa, P.: Natural Language Processing (almost) from Scratch, arXiv:1103.0398v1 (2011).
- [4] Conneau, A., Kiela, D. and Schwenk, H.: Supervised Learning of Universal Sentence Representations from Natural Language Inference Data, arXiv:1705.02364v5 (2018).
- [5] Do, Q.T.N., Bethard, S. and Moens, M.-F.: Domain Adaptation in Semantic Role Labeling Using a Neural Language Model and Linguistic Resources, *IEEE/ACM Trans. Audio, Speech, and Language Processing*, Vol.23, No.11, pp.1812–1823 (2015).
- [6] Dozat, T. and Manning, C.D.: Deep Biaffine Attention for Neural Dependency Parsing, *Proc. International Conference on Learning Representations* (2017).
- [7] Fürstena, H. and Lapata, M.: Semi-Supervised Semantic Role Labeling via Structural Alignment, *Computational Linguistics*, Vol.38, No.1, pp.135–171 (2012).
- [8] Gildea, D. and Jurafsky, D.: Automatic Labeling of Semantic Roles, *Computational Linguistics*, Vol.28, No.3, pp.1–45 (2002).
- [9] Hajič, J., Ciaramita, M., Johansson, R., Kawahara, D., Martí, M.A., Màrquez, L., Meyers, A., Nivre, J., Padó, S., Štěpánek, J., Straňák, P., Surdeanu, M., Xue, N. and Zhang, Y.: The CoNLL-2009 Shared Task: Syntactic and Semantic Dependencies in Multiple Languages, *Proc. 13th Conference on Computational Natural Language Learning (CoNLL): Shared Task*, pp.1–18 (2009).
- [10] He, L., Lee, K., Levy, O. and Zettlemoyer, L.: Jointly Predicting Predicates and Arguments in Neural Semantic Role Labeling, *Proc. 56th Annual Meeting of the Association for Computational Linguistics*, pp.364–369 (2017).
- [11] He, L., Lee, K., Lewis, M. and Zettlemoyer, L.: Deep Semantic Role Labeling: What Works and What's Next, *Proc. 55th Annual Meeting of the Association for Computational Linguistics*, pp.473–483 (2017).
- [12] Kawahara, D., Kurohashi, S. and Hasida, K.: Construction of a Japanese relevance-tagged Corpus, *Proc. 3rd International Conference on Language Resources and Evaluation*, pp.2008–2013 (2002).
- [13] Kim, Y.: Convolutional Neural Networks for Sentence Classification, *Proc. 2014 Conference on Empirical Methods in Natural Language Processing*, pp.1746–1751 (2014).
- [14] Kingsbury, P., Palmer, M. and Marcus, M.: Adding Semantic Annotation to the Penn TreeBank, *Proc. Human Language Technology Conference* (2002).
- [15] Krizhevsky, A., Sutskever, I. and Hinton, G.E.: ImageNet Classification with Deep Convolutional Neural Networks, *Proc. 25th International Conference on Neural Information Processing Systems*, pp.1097–1105 (2012).
- [16] LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W. and Jackel, L.D.: Backpropagation Applied to Handwritten Zip Code Recognition, *Neural Computation*, Vol.1, No.4, pp.541–551 (1989).
- [17] Maekawa, K., Yamazaki, M., Ogiso, T., Maruyama, T., Ogura, H., Kashino, W., Koiso, H., Yamaguchi, M., Tanaka, M. and Den, Y.: Balanced Corpus of Contemporary Written Japanese, *Language Resources and Evaluation*, Vol.48, pp.345–371 (2014).
- [18] Matsubayashi, Y. and Inui, K.: Distance-Free Modeling of Multi-Predicate Interactions in End-to-End Japanese Predicate-Argument Structure Analysis, *Proc. 27th International Conference on Computational Linguistics*, pp.94–106 (2018).
- [19] Nair, V. and Hinton, G.E.: Rectified Linear Units Improve Restricted Boltzmann Machines, *Proc. 27th International Conference on International Conference on Machine Learning*, pp.807–814 (2010).
- [20] Ouchi, H., Shindo, H., Duh, K. and Matsumoto, Y.: Joint Case Argument Identification for Japanese Predicate Argument Structure Analysis, *Proc. 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp.961–970 (2015).
- [21] Ouchi, H., Shindo, H. and Matsumoto, Y.: Neural Modeling of Multi-Predicate Interactions for Japanese Predicate Argument Structure Analysis, *Proc. 55th Annual Meeting of the Association for Computational Linguistics*, pp.1591–1600 (2017).
- [22] Ouchi, H., Shindo, H. and Matsumoto, Y.: A Span Selection Model for Semantic Role Labeling, *Proc. 2018 Conference on Empirical Methods in Natural Language Processing*, pp.1630–1642 (2018).
- [23] Palmer, M., Gildea, D. and Kingsbury, P.: The Proposition Bank: An Annotated Corpus of Semantic Roles, *Computational Linguistics*, Vol.31, No.1, pp.71–105

- (2005).
- [24] Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. and Zettlemoyer, L.: Deep contextualized word representations, *Proc. NAACL-HLT 2018*, pp.2227–2237 (2018).
 - [25] Pradhan, S., Hacioglu, K., Ward, W., Martin, J.H. and Jurafsky, D.: Semantic Role Chunking Combining Complementary Syntactic Views, *Proc. 9th Conference on Computational Natural Language Learning*, pp.217–220 (2005).
 - [26] Pradhan, S., Moschitti, A., Xue, N., Uryupina, O. and Zhang, Y.: CoNLL-2012 Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes, *Proc. Joint Conference on EMNLP and CoNLL: Shared Task*, pp.1–40 (2012).
 - [27] Shibata, T., Kawahara, D. and Kurohashi, S.: Neural Network-Based Model for Japanese Predicate Argument Structure Analysis, *Proc. 54th Annual Meeting of the Association for Computational Linguistics*, pp.1235–1244 (2016).
 - [28] Strubell, E., Verga, P., Andor, D., Weiss, D. and McCallum, A.: Linguistically-Informed Self-Attention for Semantic Role Labeling, *Proc. 2018 Conference on Empirical Methods in Natural Language Processing*, pp.5027–5038 (2018).
 - [29] Tan, Z., Wang, M., Xie, J., Chen, Y. and Shi, X.: Deep Semantic Role Labeling with Self-Attention, *Proc. AAAI*, pp.4929–4936 (2018).
 - [30] Yang, H. and Zong, C.: Multi-Predicate Semantic Role Labeling, *Proc. 2014 Conference on Empirical Methods in Natural Language Processing*, pp.363–373 (2014).
 - [31] Zhou, J. and Xu, W.: End-to-end Learning of Semantic Role Labeling Using Recurrent Neural Networks, *Proc. 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp.1127–1137 (2015).
 - [32] 麻生英樹, 安田宗樹, 前田新一, 岡野原大輔, 岡谷貴之, 久保陽太郎, ボレガラダヌシカ: 深層学習 Deep Learning, 近代科学社 (2015).
 - [33] 岡谷貴之: 深層学習, 講談社 (2015).
 - [34] 竹内孔一: 語彙概念と語彙概念構造, コーパスと自然言語処理, 松本裕治, 奥村 学 (編), 朝倉書店, pp.94–113 (2017).
 - [35] 橋田浩一: GDA 意味的修飾に基づく多用途の知的コンテツツ, 人工知能学会論文誌, Vol.13, No.4, pp.528–535 (1998).
 - [36] 浅川伸一: Python で体験する深層学習, コロナ社 (2016).
 - [37] 浅原正幸, 岡 照晃: 『国語研日本語ウェブコーパス』に基づく単語の分散表現データ, 言語処理学会第 23 回年次大会, pp.94–97 (2017).
 - [38] 石原靖弘, 竹内孔一: 係り元の末尾表現に着目した Hierarchical Tag Context Tree を利用した日本語意味役割付与システムの構築, 情報処理学会論文誌, Vol.57, pp.1611–1626 (2016).
 - [39] 飯田 龍, 小町 守, 井之上直也, 乾健太郎, 松本裕治: 述語項構造と照応関係のアノテーション: NAIST テキストコーパス構築の経験から, 自然言語処理, Vol.17, No.2, pp.25–50 (2010).



岡村 拓哉

2018 年岡山大学工学部情報系学科卒業。2018 年同大学大学院博士前期課程入学。述語項構造解析の研究に従事。



竹内 孔一 (正会員)

1998 年奈良先端科学技術大学院大学博士後期課程修了。博士 (工学)。同年学術情報センター助手。2000 年国立情報学研究所助手。2003 年岡山大学工学部情報工学科講師。2005 年同大学大学院講師。現在に至る。主に、専門用語研究、述語項構造の言語資源構築と解析に従事。言語処理学会, 人工知能学会, 電子情報通信学会, ACM 各会員。



石原 靖弘

2016 年 9 月岡山大学大学院自然科学研究科博士後期課程修了。博士 (工学)。現在, 岡山大学工学部特任助教。専門は自然言語処理。