

# マルチタスク学習を用いたニューラル文内述語項構造解析

大森 光 小町 守  
首都大学東京

omori-hikaru@ed.tmu.ac.jp, komachi@tmu.ac.jp

## 1 はじめに

述語項構造解析とはテキスト中に存在する述語とその項との意味構造を解析するタスクである。「誰が」「何を」「誰に」「どうした」のように文の構造を整理することは、機械翻訳や含意関係認識など複雑の文章の解析を必要とする処理のために有用である。述語項構造解析は動詞や形容詞といった述語を対象にし、その項を解析するタスクであるが、文中には名詞でも項を持つものが多く存在する。例えば、「報告」のようなサ変名詞や「救い」のような動詞から派生した名詞がこれにあたり、このような名詞を事態性名詞と呼ぶ。

NAIST テキストコーパス [3] では、述語と事態性名詞はどちらも必須格（ガ格、ヲ格、ニ格）の項を持つ。また Iida ら [3] によると、述語は同一文節内に項を持つことはほぼないが、事態性名詞の場合はヲ格とニ格の項の約半分が同一文節内に出現する。このことから、事態性名詞の項構造解析と述語項構造解析は関係性が高いが別のタスクであるといえる。

先行研究では、機械学習を用いた述語項構造解析の研究が盛んに行われてきた。しかしこれらのほとんどが述語のみに焦点を当てたものであり、事態性名詞を扱った研究 [5, 12] は少ない。文章の意味構造を整理し、正しく文脈解析を行う上で、述語のみを対象にした項構造解析は不十分である。そこで我々は、述語項構造解析と事態性名詞の項構造解析をマルチタスク学習するモデルを提案する。このモデルは先行研究で用いられた end-to-end な多層双方向 Recurrent Neural Network (RNN) がベースであり、入力層と RNN 層と出力層においてタスク間で共通の知識とタスク特有の知識を区別するネットワークを有する (図 1 右)。さらに、この提案モデルは先行研究と同様に述語と項の統語関係を考慮する。我々は日本語述語項構造解析において一般的なベンチマークとして使われる NAIST テキストコーパスを用いて実験を行った。

本稿の貢献は以下の 3 つである。

1. 初めてマルチタスク学習を用いて述語項構造解析と事態性名詞の項構造解析を行い、双方のタスクで解析精度が向上することを示した。
2. 統語情報を素性として加えることによって、複数の述語項関係を考慮しない単純なモデルが文内述語項構造解析において最高精度を達成した。
3. 初めて深層学習を用いて事態性名詞の項構造解析を行った。

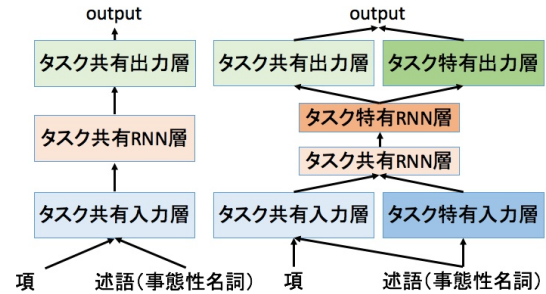


図 1: シングルタスクモデル (左), マルチタスクモデル (右)

## 2 述語と事態性名詞の項構造解析

### 2.1 タスク設定

日本語述語 (事態性名詞) 項構造解析とは、ある述語 (事態性名詞) に対する項を抽出しガ格、ヲ格、ニ格の 3 つの格ラベルを付与するタスク [3] である。項には述語 (事態性名詞) の位置によって 4 つのカテゴリ [12] に大別される。

**Dep** 項が述語 (事態性名詞) に係る、もしくは述語 (事態性名詞) が項に係る。

**Zero** ゼロ照応によって項が省略され、項と述語 (事態性名詞) が同一文内にあるが、直接係り関係にない。

**Inter-Zero** ゼロ照応によって項が省略され、項と述語 (事態性名詞) が同一文内にない。

**Bunsetsu** 項と述語 (事態性名詞) が同一文節内にある。

単語単位に分かち書きされた文  $w = w_1, w_2, \dots, w_T$  と述語 (事態性名詞) の位置  $p = p_1, p_2, \dots, p_q$  が入力として与えられる。先行研究 [2, 4, 10] は Inter-Zero の解析を行っているが、これは他のカテゴリと違い文書全体を探索しなければならないため、解析がより困難である。本研究では既存研究 [8, 9, 6, 7] に従い、項と述語 (事態性名詞) が同一文内にある事例のみを解析対象とする。また先行研究 [9, 7] にならい、Bunsetsu カテゴリに該当する項は述語項構造解析の評価対象から除外する。

### 2.2 シングルタスクモデル

我々は、先行研究 [13, 9, 7] で用いられた end-to-end な多層双方向 RNN をベースにしたマルチタスク項構造解析モデルを提案する。シングルタスクモデルを図 2 に示す。はじめに、入力層で単語  $w_t \in [w_1, \dots, w_T]$

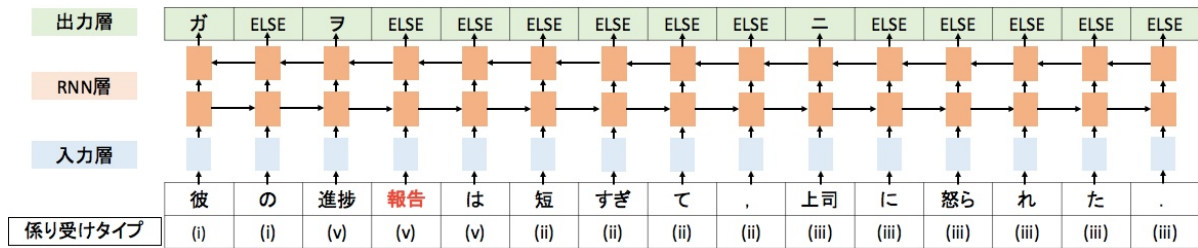


図 2: end-to-end ベースラインモデル

を素性ベクトル  $\mathbf{x}_t \in [\mathbf{x}_1, \dots, \mathbf{x}_T]$  に変換する. 次に, RNN 層に入力し  $l \in [1, \dots, L]$  層目の隠れ層ベクトル  $\mathbf{h}_t^l \in \mathbb{R}^{d_h}$  を得る. RNN 層は Stacked Gated Recurrent Unit で構成する. 最後に, 出力層に各隠れ層のベクトル  $\mathbf{h}_t^l$  を入力し, 出力ベクトル  $\mathbf{o}_t$  を得る. 出力ベクトル  $\mathbf{o}_t$  の各要素は, 項候補が述語のガ格, ヲ格, ニ格であるか, 必須格でないか (ELSE) を表す確率値となっており, 確率値最大のラベルを予測ラベル  $y_t$  として出力する.

### 2.2.1 入力素性

素性ベクトル  $\mathbf{x}_t$  は以下のように定義される.

$$\mathbf{x}_t = \mathbf{x}_t^{\text{as}} \oplus \mathbf{x}_t^{\text{posi}} \oplus \mathbf{x}_t^{\text{dep}} \oplus \mathbf{x}_t^{\text{type}} \oplus \mathbf{x}_p^{\text{task}} \quad (1)$$

$\oplus$  はベクトルの結合を表す.

**述語 (事態性名詞) と項** 述語 (事態性名詞)  $w_p$  と項候補の単語  $w_t$  は単語埋め込み行列によってベクトル  $\mathbf{x}_t^{\text{as}} \in \mathbb{R}^{2d_w}$  にそれぞれ変換される.

**位置** 述語 (事態性名詞) と単語の位置関係を示す素性である. 述語 (事態性名詞) のインデックスから単語のインデックスを引いた値を, 位置埋め込み行列を用いて位置ベクトル  $\mathbf{x}_t^{\text{posi}} \in \mathbb{R}^{d_p}$  に変換する.

**係り受け** 述語 (事態性名詞) と項候補の係り受け関係を示す素性  $d_t$  である. 係り受け関係のタイプは 5 種類に設定した:

- i). 項が述語 (事態性名詞) に係る.
- ii). 述語 (事態性名詞) が項に係る.
- iii). 項と述語 (事態性名詞) が係り受け関係がない.
- iv). 述語が項と同一文節内にある.
- v). 事態性名詞が項と同一文節内にある.

$d_t$  は係り受け関係埋め込み行列によって係り受けベクトル  $\mathbf{x}_t^{\text{dep}} \in \mathbb{R}^{d_d}$  に変換される.

先行研究でも係り受け関係を用いた素性が使用されている. Imamura ら [4] は述語と項の間に係り受け関係があるか否かの 2 値素性を用いた. 我々は事態性名詞に適応させるために, より細かく係り受け関係を分類した. Matsubayashi & Inui [6] は統語関係埋め込みベクトルを用いて述語項間の関係を表現した. 対照的に, 我々は述語と事態性名詞を区別するためにそれぞれで異なる素性を定義し, これらの潜在的な構造を見つけるように学習を行った.

**事態性** 文中の全ての述語 (事態性名詞) の位置にフラグを立てる 2 値素性ベクトル  $\mathbf{x}_t^{\text{type}}$  である. この素性は Matsubayashi & Inui [7] に影響されたものである. この素性の目的は述語が項になるのを防ぎ, 一部の事態性名詞が項になるのを助けることである.

**タスクラベル** 述語項構造解析なら 1, 事態性の項構造解析なら 0 を示す 2 値素性ベクトル  $\mathbf{x}_p^{\text{task}}$  である.

## 3 マルチタスクモデル

### 3.1 マルチ入力層

表層形が同じであっても, 述語と事態性名詞では文脈が異なることを考慮するために, 我々はタスク独自の単語ベクトルを学習するタスク特有単語埋め込み行列を用意する.

述語 (事態性名詞)  $w_p$  はタスク特有単語埋め込み行列によってベクトル  $\mathbf{x}_t^p, \mathbf{x}_t^s \in \mathbb{R}^{d_w}$  に変換される. 素性ベクトル  $\bar{\mathbf{x}}_t$  は以下のように定義される.

$$\bar{\mathbf{x}}_t = \begin{cases} \mathbf{x}_t \oplus \mathbf{x}_t^p & (\text{PASA}) \\ \mathbf{x}_t \oplus \mathbf{x}_t^s & (\text{ENASA}) \end{cases} \quad (2)$$

PASA は述語項構造解析を, ENASA は事態性名詞の項構造解析を示す.

### 3.2 マルチ RNN 層

先行研究 [11, 1] では, 容易なタスクから得られた特徴を困難なタスクに活用する階層的なマルチタスク学習のモデルが提案されている. そこで我々はタスク共有の RNN の上にタスク特有の RNN を階層的に重ねるネットワークを構築する. 共有の RNN でタスク共有の知識表現を学習し, タスク特有の RNN でそれぞれのタスクに調整する. 各タイムステップ  $t$  において,  $l' \in [1, \dots, L']$  層目の隠れ層ベクトル  $\mathbf{m}_t^{l'} \in \mathbb{R}^{d_h}$  は以下のように計算される.

$$\mathbf{m}_t^{l'} = \begin{cases} g^{l'}(\mathbf{m}_t^{l'-1}, \mathbf{m}_{t-1}^{l'-1}) & (l' = \text{odd}) \\ g^{l'}(\mathbf{m}_t^{l'-1}, \mathbf{m}_{t+1}^{l'-1}) & (l' = \text{even}) \end{cases} \quad (3)$$

$$g^{l'}(\cdot) = \begin{cases} g_p^{l'}(\cdot) & (\text{PASA}) \\ g_n^{l'}(\cdot) & (\text{ENASA}) \end{cases} \quad (4)$$

$g^{l'}(\cdot), g_p^{l'}(\cdot), g_n^{l'}(\cdot)$  は  $l'$  層目の GRU の処理を表す. また  $\mathbf{m}_t^0 = \mathbf{h}_t^L$  である.

### 3.3 マルチ出力層

述語と事態性名詞では項の取り方が異なるため、出力層をタスク共有の層とタスク特有の層に分割した。タスク特有の出力ベクトルは以下のように計算される。

$$\mathbf{o}_t^p = \mathbf{W}_o^p \mathbf{h}_t + \mathbf{b}_o^p \quad (5)$$

$$\mathbf{o}_t^n = \mathbf{W}_o^n \mathbf{h}_t + \mathbf{b}_o^n \quad (6)$$

$$\mathbf{g}_t = \sigma(\mathbf{W}_g \mathbf{h}_t + \mathbf{b}_g) \quad (7)$$

$\mathbf{h}_t$  は RNN の最終層の各隠れ層のベクトルである。タスク特有の出力ベクトル  $\mathbf{o}_t^p, \mathbf{o}_t^n$  はゲート  $\mathbf{g}_t$  によってタスク共有の出力ベクトル  $\mathbf{o}_t$  と結合する。

$$\mathbf{c}_t = \begin{cases} \mathbf{g}_t \odot \mathbf{o}_t + (1 - \mathbf{g}_t) \odot \mathbf{o}_t^p & (\text{PASA}) \\ \mathbf{g}_t \odot \mathbf{o}_t + (1 - \mathbf{g}_t) \odot \mathbf{o}_t^n & (\text{ENASA}) \end{cases} \quad (8)$$

$$\bar{\mathbf{o}}_t = \text{softmax}(\mathbf{c}_t) \quad (9)$$

$\odot$  は要素積を表す。出力ベクトル  $\bar{\mathbf{o}}_t$  は、[ガ格, ヲ格, 二格, ELSE] の確率を表すベクトルである。

## 4 実験

### 4.1 実験設定

実験データには NAIST テキストコーパス 1.5 [3] を使い、Taira ら [12] と同様の設定で学習、開発、評価データに分割した。

開発・評価データについて、同じ必須格の項が文中に2つ以上ある場合の正解データの作り方は、述語と係り受けの関係にある項のみを正解とし、それ以外の項は ELSE ラベルにラベル付けする。係り受けの関係にある項がない場合は距離  $|w_p - w_t|$  が最短の項を正解とする。距離が等しい場合は述語より左側の項を正解とする。

NAIST テキストコーパス 1.5 では、“サ変接続名詞+スル”のような述語があった場合に“スル”を述語単語としてアノテーションしている。述語と事態性名詞の表層形を一致させ、効果的に学習を行うために、我々はこの場合“サ変接続名詞”を述語単語として扱うように前処理を行った。例えば述語“報告する”と事態性名詞“報告”があった場合、前処理前の  $w_p$  はそれぞれ“する”、“報告”であるが、前処理後は“報告”に統一される。

### 4.2 ハイパーパラメータ

単語分散表現の初期値として朝日新聞単語ベクトル<sup>1</sup>の事前学習されたベクトルを使用した。最適化手法には AdaDelta を適用した。エポック数は 20 に設定し、開発データの F 値が最も高かったエポックのモデルで評価データを評価する。ハイパーパラメータを表 1 に示す。

### 4.3 実験結果

NAIST テキストコーパス 1.5 の評価データで各モデルの評価を行った。文内述語項構造解析と文内事態性名詞の項構造解析の実験結果を表 2, 3 に示す。

単語の埋め込みベクトルの次元数 $d_w$	300
位置埋め込みベクトルの次元数 $d_p$	32
係り受け関係埋め込みベクトルの次元数 $d_d$	16
GRU の隠れ層の次元数 $d_h$	300
GRU の層数 $L$	4
タスク特有単語の埋め込みベクトルの次元数 $d_w'$	16
タスク特有 GRU の隠れ層の次元数 $d_h'$	300
タスク特有 GRU の層数 $L'$	2
dropout rate	0.4
batch size	8
gradient clipping	4

表 1: ハイパーパラメータ

**述語項構造解析** 2 行目の提案モデルでは述語と項の係り受け関係を示す素性はいっていない。Multi-RNN と Multi-ALL は Single と比較して、それぞれ 0.29, 0.2 ポイントの精度向上を示した。Ouchi ら [9], Matsubayashi & Inui [7] は複数の述語項関係を考慮することで、全体の F 値でそれぞれ 0.27, 0.55 ポイントの精度向上を示している。このことから事態性名詞の項構造解析のマルチタスク学習がこれらの研究と同程度の効果を持つことを示すことができた。

3 行目に係り受け素性を含めた全ての素性を用いた提案手法のモデルの結果をまとめた。Multi-ALL+DEP が先行研究を含めた全てのモデルの中で最も高い解析精度を示した。特に Dep の事例において係り受け素性の効果が高いことが分かった。一方で Zero の事例においては解析精度が低く、事例数の多い Dep を優先した最適化を行っていることが分かった。

4 行目にアンサンブルしたモデルの結果をまとめた。最終的に、我々の提案モデルは先行研究を 0.67 ポイント上回る解析精度を示した。

またこのモデルは文内の述語の項推定を独立に行う単純なモデルであるが、先行研究 [9, 7] より高い解析精度を示している。近年では、複数の述語を同時に考慮し文内単位で最適化を行う手法が盛んに研究されてきたが、end-to-end モデルにおける統語情報の活用方法も今後研究されるべき課題の一つである。

**事態性名詞の項構造解析** Taira+08 は NAIST テキストコーパス 1.4 での実験結果であるため、提案手法との厳密な比較はできない。Single と Multi-ALL を比較すると全体の F 値が向上したが、一方述語項構造解析で最高精度を示した Multi-ALL+DEP は Multi-ALL を下回る精度を示した。このことから、項述語間の係り受け関係は事態性名詞の項構造解析において有効な素性でないことが分かった。

### 4.4 事例分析

提案モデルの効果を分析するために、評価データにおいて Multi-RNN モデルでは正しく予測できたが、Single モデルでは誤った予測を出力した事例を表 4 に示す。述語“結成する”に対して、“会長”がガ格、“会派”がヲ格である。しかしながら、Single モデルはどちらの格もこの文中に存在しないと間違えて出力している。各タスクの学習データを比較すると、文内事態性名詞の項構造解析の事例数は文内述語項構造解析の事例数の約 3 分の 1 ほどであるにも関わらず、事態性名詞“結成”を対象にした事例数は述語“結成する”を

<sup>1</sup>[http://www.asahi.com/shimbun/medialab/word\\_embedding](http://www.asahi.com/shimbun/medialab/word_embedding)

Method			Dep				Zero			
	ALL	SD	ALL	ガ	ヲ	ニ	ALL	ガ	ヲ	ニ
Ouchi+ 17 [9]	81.42		88.17	88.75	93.68	64.38	47.12	50.65	32.35	7.52
M&I 17 [6]	83.50	±0.17	89.89	91.19	95.18	61.90	51.79	54.69	41.8	17
M&I 18 [7]	83.94	±0.12	90.26	90.88	94.99	67.57	<b>55.55</b>	<b>57.99</b>	<b>48.9</b>	<b>23</b>
Single	83.62	±0.17	90.09	90.45	94.84	69.77	51.87	54.73	43.48	11.40
Multi-RNN	83.91	±0.23	90.17	90.58	95.07	67.94	53.31	55.85	45.71	9.97
Multi-ALL	83.82	±0.10	90.15	90.68	95.06	67.56	53.50	56.37	45.36	8.70
Multi-ALL+DEP	<b>84.75</b>	±0.16	<b>90.88</b>	<b>91.40</b>	<b>95.37</b>	<b>71.02</b>	52.35	55.10	42.54	2.32
M&I 17 (ens. of 5)	84.07		90.24	91.59	95.29	62.61	53.66	56.47	44.7	16
M&I 18 (ens. of 10)	85.34		91.26	91.84	95.57	70.8	<b>58.07</b>	<b>60.21</b>	<b>52.5</b>	<b>26</b>
Multi-ALL+DEP (ens. of 5)	<b>86.01</b>		<b>91.63</b>	<b>92.15</b>	<b>95.80</b>	<b>72.95</b>	54.99	57.84	45.20	0

表 2: 評価データにおける述語項構造解析の F1 スコア. Single はシングルタスクモデルを, Multi-ALL はマルチタスクモデルを, Multi-RNN は RNN 層のみをマルチ RNN 層に拡張したモデルを示す.

Method			Dep				Zero				Bunsetsu			
	ALL	SD	ALL	ガ	ヲ	ニ	ALL	ガ	ヲ	ニ	ALL	ガ	ヲ	ニ
Taira+ 08 [12] on NTC 1.4			68.01	62.46	56.05		36.19	20.46	6.62		78.93	77.96	58.13	
Single	66.21	±0.15	74.64	76.06	74.54	51.28	46.05	49.67	33.36	13.63	78.24	76.67	81.75	48.55
Multi-ALL	<b>68.00</b>	±0.41	75.90	77.16	76.05	53.00	<b>49.66</b>	<b>53.37</b>	<b>37.64</b>	14.46	79.05	77.32	82.61	<b>51.83</b>
Multi-ALL+DEP	67.68	±0.39	<b>75.95</b>	<b>77.18</b>	<b>76.11</b>	<b>55.26</b>	47.57	51.21	35.14	<b>15.65</b>	<b>79.06</b>	<b>77.44</b>	<b>82.66</b>	51.10
Multi-ALL (ens. of 5)	<b>71.14</b>		<b>78.63</b>	<b>79.66</b>	<b>78.83</b>	58.29	<b>52.49</b>	<b>56.41</b>	<b>39.02</b>	16.42	<b>81.90</b>	<b>80.25</b>	<b>85.21</b>	<b>56.29</b>
Multi-ALL+DEP (ens. of 5)	69.90		77.86	78.89	78.16	<b>58.46</b>	49.36	53.10	36.36	<b>17.23</b>	81.16	79.74	84.57	52.99

表 3: 評価データにおける事態性名詞の項構造解析の F1 スコア

社会党の山花真夫・新民主連合会長は十七日午前、新たに結成する新会派に参加する同党所属国会議員...国対委員長に提出した。		
Multi-RNN		Single
ガ格	会長	-
ヲ格	会派	-
ニ格	-	-

表 4: 評価データにおける述語項構造解析の解析事例

対象にした事例数の約 2 倍であった。このことから、Multi-RNN モデルはマルチタスク学習によって効果的に事態性名詞の情報を活用できていることが分かる。

## 5 おわりに

本稿で我々は、述語項構造解析と事態性名詞の項構造解析をマルチタスク学習するモデルを提案した。その結果 NAIST テキストコーパスにおいて双方のタスクで解析精度が向上することを示した。そして、係り受け関係を素性として加えることによって、文内述語項構造解析において最高精度を達成した。また、本研究は深層学習を用いて事態性名詞の項構造解析を行った初めての研究である。今後の展望としては、複数の述語と事態性名詞を考慮したモデルへの改良などが挙げられる。

## 参考文献

- [1] Kazuma Hashimoto, caiming xiong, Yoshimasa Tsuruoka, and Richard Socher. A joint many-task model: Growing a neural network for multiple NLP tasks. In *EMNLP*, pp. 1923–1933, 2017.
- [2] Ryu Iida, Kentaro Inui, and Yuji Matsumoto. Exploiting syntactic patterns as clues in zero-anaphora resolution. In *COLING/ACL*, pp. 625–632, 2006.

- [3] Ryu Iida, Mamoru Komachi, Kentaro Inui, and Yuji Matsumoto. Annotating a Japanese text corpus with predicate-argument and coreference relations. In *Linguistic Annotation Workshop*, pp. 132–139, 2007.
- [4] Kenji Imamura, Kuniko Saito, and Tomoko Izumi. Discriminative approach to predicate-argument structure analysis with zero-anaphora resolution. In *ACL-IJCNLP*, pp. 85–88, 2009.
- [5] Mamoru Komachi, Ryu Iida, Kentaro Inui, and Yuji Matsumoto. Learning based argument structure analysis of event-nouns in Japanese. In *PACLING*, pp. 120–128, 2007.
- [6] Yuichiroh Matsubayashi and Kentaro Inui. Revisiting the design issues of local models for Japanese predicate-argument structure analysis. In *IJCNLP*, pp. 128–133, 2017.
- [7] Yuichiroh Matsubayashi and Kentaro Inui. Distance-free modeling of multi-predicate interactions in end-to-end Japanese predicate argument structure analysis. In *COLING*, pp. 94–106, 2018.
- [8] Hiroki Ouchi, Hiroyuki Shindo, Kevin Duh, and Yuji Matsumoto. Joint case argument identification for Japanese predicate argument structure analysis. In *ACL-IJCNLP*, pp. 961–970, 2015.
- [9] Hiroki Ouchi, Hiroyuki Shindo, and Yuji Matsumoto. Neural modeling of multi-predicate interactions for Japanese predicate argument structure analysis. In *ACL*, pp. 1591–1600, 2017.
- [10] Ryohei Sasano and Sadao Kurohashi. A discriminative approach to Japanese zero anaphora resolution with large-scale lexicalized case frames. In *IJCNLP*, pp. 758–766, 2011.
- [11] Anders Søgaard and Yoav Goldberg. Deep multi-task learning with low level tasks supervised at lower layers. In *ACL*, pp. 231–235, 2016.
- [12] Hiroto Taira, Sanae Fujita, and Masaaki Nagata. A Japanese predicate argument structure analysis using decision lists. In *EMNLP*, pp. 523–532, 2008.
- [13] Jie Zhou and Wei Xu. End-to-end learning of semantic role labeling using recurrent neural networks. In *ACL-IJCNLP*, pp. 1127–1137, 2015.