Vol.2017-NL-233 No.12
2017/10/25

意味役割付与における未知分野へのニューラル分野適応技術

大内 啓樹[1,a)]　進藤 裕之[1,b)]　松本 裕治[1,c)]

概要：意味役割付与 (Semantic Role Labeling) において，学習データと分野の異なるデータを解析する際，解析性能が低下するという問題が知られている．このような問題に対処するため，分野適応技術に関する研究が行われてきた．多くの既存研究における分野適応の問題設定は，解析対象となるデータの分野は既知のものとして扱っている．しかし，現実的な解析を考えると，必ずしも解析対象データの分野が自明であるわけではない．そこで本研究では，解析対象データの分野が未知の場合の分野適応 (未知分野適応) に取り組む．具体的には，未知分野適応として 2 つの問題設定を定式化し，それらに有用なモデリングフレームワークを提案する．CoNLL-2012 Shared Task のデータを用いた評価実験を行い，提案フレームワークによるモデルが未知分野に対しても頑健に解析可能であることを確認した．また，解析結果の誤り分析から，意味役割ラベルの予測に大きな改善の余地が残されていることがわかった．

1. はじめに

意味役割付与は，「いつ，どこで，誰が，何を，誰に，どうした」といった述語と項の関係を同定する意味解析タスクである．近年，ニューラルネットワークを用いた End-to-End 型の手法によって，構文情報を使用せずとも従来法を上回る解析精度が報告されている [20], [29], [43]．しかし，学習データと異なる分野のデータを解析する際に，精度が低下するという問題があり，分野適応技術開発の必要性が指摘されている [20], [42]．

Yang ら [42] は，意味役割付与における分野適応研究に取り組んでいる．彼らは，元分野 (Source Domain) として新聞データを用いて，目標分野 (Target Domain) である小説データへの適応を試みた．Deep Belief Networks を用いることにより，解析精度の向上を実現している．同様の問題設定で，WordNet[30] などの外部資源やニューラル言語モデルを用いて，目標分野に適応する手法も考案されている [13]．

これらの研究を踏まえ，我々は，意味役割付与における分野適応として取り組むべき課題が 2 つあると考える．

1 つ目の課題は，**未知の分野への適応**である．これまでの意味役割付与における分野適応研究では，解析対象の分野 (目標分野) が既知の設定で分野適応が行われてきた．例えば，目標分野が小説であることはわかっており，いかにして小説に適応するかを考える．しかし，実応用の場では目標分野が未知の場合も多い．Google 翻訳[*1]などのアプリケーションでは，解析対象テキストの分野は未知である．したがって，解析したいテキストの分野が未知であっても，頑健に解析可能な適応技術が必要となる．

2 つ目の課題は，**複数元分野データを用いた適応**である．言語処理における多くの既存研究では，単一分野の教師ラベル付きデータを用いて分野適応を行ってきた．例えば，大規模な新聞記事 (Wall Street Journal) をモデル学習の軸として，小説 (Brown Corpus の小説セクション) に適応することを考える．しかし，現在は，OntoNotes[21], [40] に代表されるように，複数の異なる分野のテキストからなる教師ラベル付きデータが入手可能となっている．したがって，そのようなデータにおける分野間の違いを考慮した学習手法を研究することにより，より良いモデルの構築が可能になると期待できる．

本研究では，上記 2 つの課題に同時に取り組む．まず，1 つ目の課題である未知分野適応として，2 つの異なる問題設定を定式化する．具体的には，未知の目標分野と同一分野のデータを学習データに含む設定 (Target-Covered 未知分野適応) と，同一分野のデータを学習データに含まない設定 (Zero-Shot 未知分野適応) である．次に，2 つ目の課題である複数元分野データを用いた分野適応として，複数分野を考慮して頑健に未知分野を解析可能なモデリングフレームワークを提案する．具体的には，次の 2 つのフレームワークを提案する: (1) 単一エキスパート選択モデ

[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://translate.google.com/m/translate

リング，(2) 複数エキスパート統合モデリング．両フレームワークとも，K 個の分野それぞれに特化したモデル (分野エキスパート; Domain Expert) を利用し，未知分野に動的に適応する．単一エキスパート選択モデリングでは，入力文が与えられた際に，その文を解析するのに適した 1 つのエキスパートを選択して予測を行う．複数エキスパート統合モデリングでは，入力文をすべてのエキスパートで解析し，それらの解析結果を統合して最終的な予測を行う．

CoNLL-2012 Shared Task のデータセット [32] を用いた評価実験で，提案フレームワークからインスタンス化したモデルの有効性を確認した．また，解析結果の誤り分析から，意味役割ラベルの予測に大きな改善の余地が残されていることがわかった．

本研究の貢献は以下の 3 点に要約できる．

- 意味役割付与における未知分野適応の問題設定の定式化．
- 複数元分野を考慮した未知分野適応手法の提案とその有用性の評価．
- 分野別の誤り傾向の分析と今後の手法の改善点の示唆．

2. 意味役割付与における分野適応

2.1 意味役割付与の問題設定

意味役割付与では，解析対象の 1 つの述語に対して，項とその意味役割を予測する．近年では，ニューラルネットワークを用いて，意味役割付与を系列ラベリング問題 (Sequence Labeling) として解析を行い，従来手法を上回る結果が報告されている [20], [43]．本研究でもこれらの研究に基づき，意味役割付与を系列ラベリング問題として解く．

定式的には，解析対象の述語 $p \in \mathcal{P}$ とその述語を含む文 (単語列) $\boldsymbol{x} = (x_1, \ldots, x_T)$ を入力とし，スコア最大のラベル列 $\boldsymbol{y} = (y_1, \ldots, y_T)$ を出力する．

$$\hat{\boldsymbol{y}} = \underset{\boldsymbol{y} \in \mathcal{Y}}{\operatorname{argmax}} f(\boldsymbol{y}|\boldsymbol{x}, p)$$

ここで，$\mathcal{Y}$ はすべての可能なラベル列の集合を表す．それぞれのラベル y_t は BIO タグセット $\mathcal{T}$ に属する．学習データとして $\mathcal{D} = \{(\boldsymbol{x}_i, p_i, \boldsymbol{y}_i)\}_1^N$ が与えられ，関数 $f : \mathcal{X} \times \mathcal{P} \rightarrow \mathcal{Y}$ を学習する．

2.2 分野適応の問題設定

分野適応は，使用する学習データにおける教師ラベルの有無によって，「教師あり/半教師あり/教師なし」分野適応の 3 つに分類される．また，単一分野 (Single-Souce) からなる学習データと複数分野 (Multi-Source) からなる学習データのどちらを利用するかによって，「単一元/複数元」分野適応に分類される．本研究では「教師あり/複数元」分野適応に取り組む．

定式的には，各元分野 (Source Domain) $k \in [1, K]$ に対して，学習データ $\mathcal{D}_k^{sc} = \{(x_i, y_i)\}_1^{N^{(k)}}$ が与えられている．また，目標分野 (Target Domain) の学習データ $\mathcal{D}^{tg} = \{(x_i, y_i)\}_1^{N^{tg}}$ も与えられている．ここで，$x_i \in \mathcal{X}$ は入力，$y_i \in \mathcal{Y}$ は出力を表す．これらの学習データ $\{\mathcal{D}_k^{sc}\}_1^K \cup \mathcal{D}^{tg}$ を用いて関数 $f : \mathcal{X} \rightarrow \mathcal{Y}$ を学習する．

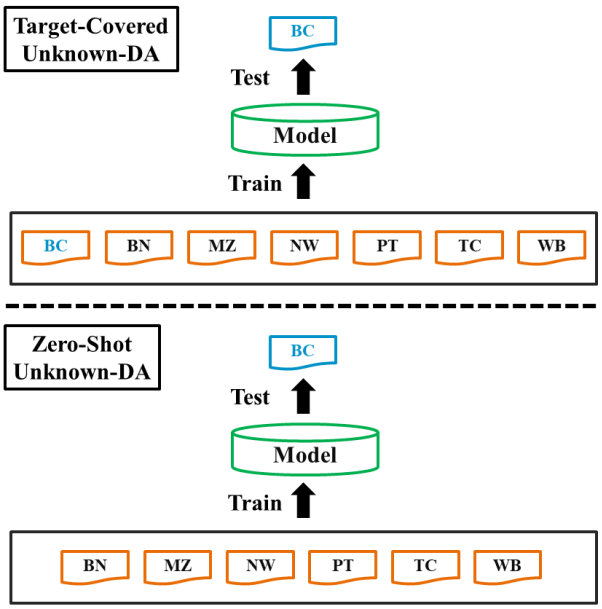

図 1 Target-Covered/Zero-Shot 未知分野適応の概略図．Target-Covered の設定は学習データに目標分野を含み，Zero-Shot の設定は学習データに目標分野を含まない．

2.3 意味役割付与における分野適応の問題設定

これまでの意味役割付与において，単一元 (Single-Source) 分野適応が研究されてきた [13], [42]．本節では，意味役割付与タスクでの単一元分野適応の問題設定を記述する．

元分野の学習データ $\mathcal{D}^{sc} = \{(\boldsymbol{x}_i, p_i, \boldsymbol{y}_i)\}_1^{N^{sc}}$ が与えられる．教師あり分野適応なら，目標分野の教師ラベル付き学習データ $\mathcal{D}^{tg} = \{(\boldsymbol{x}_i, p_i, \boldsymbol{y}_i)\}_1^{N^{tg}}$ が与えられる．一方，教師なし分野適応なら，目標分野の教師ラベルなし学習データ $\mathcal{D}^{tg} = \{(\boldsymbol{x}_i, p_i)\}_1^{N^{tg}}$ が与えられる．これらのデータを用いて関数 $f : \mathcal{X} \times \mathcal{P} \rightarrow \mathcal{Y}$ を学習する．

3. 未知分野適応

一般的な分野適応の設定では目標分野が既知である (2.2 節)．しかし，実応用の場では目標分野が未知の場合も多い．そこで本研究では，未知の目標分野への適応 (**未知分野適応; Unknown-Domain Adaptation**) に取り組む．具体的には，未知分野適応を以下の 2 つの問題設定に細分化し，それぞれに取り組む: (1) **Target-Covered 未知分野適応** (Target-Covered Unknown-Domain Adaptation)，(2) **Zero-Shot 未知分野適応** (Zero-Shot Unknown-Domain Adaptation)．以降の節で各設定を詳しく記述する．

3.1 Target-Covered 未知分野適応

図 1 の上図は Target-Covered 未知分野適応の概略を示している．問題設定として，**未知の目標分野が学習データに含まれている**状況を想定する．例えば，目標分野が BC である場合，学習データに目標分野 BC の事例が含まれている．また，本研究で取り組むのは未知分野適応であるため，目標分野がどの分野かは未知である．したがって，学習データとして目標分野の事例が含まれてはいるが，どの分野が目標分野であるかはわからない状況を想定している．

具体的には，評価時は，未知目標分野 unk の文と解析対象述語のペア $(\boldsymbol{x}, p) \sim \mathcal{D}_{unk}$ を受け取り，ラベル $\hat{\boldsymbol{y}}$ を出力する．学習時は，各元分野 $k \in [1, K]$ に対して学習データ $\mathcal{D}_k = \{(\boldsymbol{x}_i, p_i, \boldsymbol{y}_i)\}_1^{N^{(k)}}$ が与えられる．ここで，評価時の目標分野 unk は未知であるが，$unk \in [1, K]$ であるため，学習データに目標分野の事例が含まれる．これらの学習データ $\{\mathcal{D}_k\}_1^K$ を用いて，関数 $f : \mathcal{X} \times \mathcal{P} \to \mathcal{Y}$ を学習する．

3.2 Zero-Shot 未知分野適応

図 1 の下図は Zero-Shot 未知分野適応の概略を示している．問題設定として，**未知の目標分野が学習データに含まれていない**状況を想定する．例えば，目標分野が BC である場合，学習データに目標分野 BC の事例は含まれていない．つまり，目標分野以外の学習データの事例をどのように用いて適応するかがポイントとなる．したがって，Target-Covered 未知分野適応よりも困難な問題であると言える．

具体的には，評価時は，未知目標分野 unk の文と解析対象述語のペア $(\boldsymbol{x}, p) \sim \mathcal{D}_{unk}$ を受け取り，ラベル $\hat{\boldsymbol{y}}$ を出力する．学習時は，各元分野 $k \in [1, K]$ に対して学習データ $\mathcal{D}_k = \{(\boldsymbol{x}_i, p_i, \boldsymbol{y}_i)\}_1^{N^{(k)}}$ が与えられる．ここで，評価時の目標分野 unk は未知であり，$unk \notin [1, K]$ であるため，学習データに目標分野の事例は含まれない．これらの学習データ $\{\mathcal{D}_k\}_1^K$ を用いて，関数 $f : \mathcal{X} \times \mathcal{P} \to \mathcal{Y}$ を学習する．

4. 手法

本研究では，Mixture of Experts (MoE)[23], [24], [36] をベースとした分野適応手法を提案する．本手法は，各分野の解析に特化したモデル (Domain Expert; DE) を組み合わせて，最終的な予測を行う．その際，各分野のエキスパートをどのように最終的な予測に反映させるかは自明ではない．

そこで我々は，2 つの異なるモデリングフレームワークを提案する．1 つ目は，入力文に基づいて，1 つのエキスパートを動的に選択して予測を行うフレームワークである**単一エキスパート選択モデリング**である．2 つ目は，入力文に基づいて，すべてのエキスパートを統合して予測を行うフレームワークである**複数エキスパート統合モデリング**である．

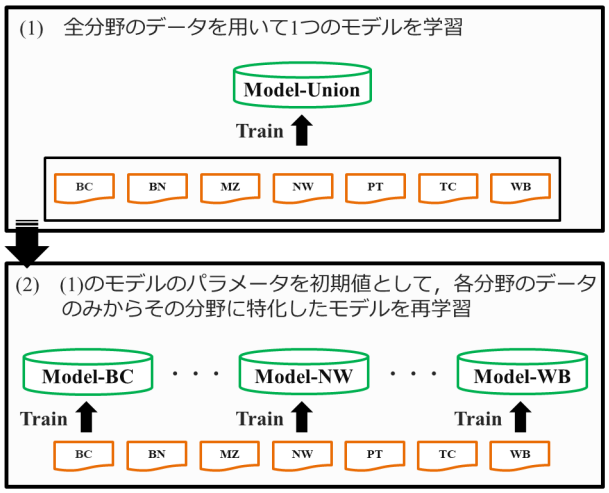

図 2 各分野に特化したモデル (エキスパート) の構築手順.

本章では，まず最初に，両方のフレームワークの共通基盤について述べる．次に，各モデリングフレームワークの詳細について述べる．

4.1 フレームワークの共通基盤

本フレームワークでは，複数の分野エキスパートを用いる．図 2 が示すように，これらの分野エキスパートは以下のような手順で構築される．

(1) 与えられた全分野の学習データから，特定の分野に依存しない 1 つのモデル (Union Model) を学習する．
(2) (1) で構築したモデルのパラメータを初期値として，各分野の学習データのみからその分野に特化したモデルを再学習 (Fine-Tuning) する．

各モデルは，双方向リカレントニューラルネットワーク (Bidirectional Recurrent Neural Networks)[18], [35] を多層にしたモデル [20], [43] に基づいている．モデルの詳細は付録 A.1 に記す．

4.2 単一エキスパート選択モデリング

単一エキスパート選択モデリングのインスタンス化として，どのエキスパートを選択すべきかを判断する分類器を用いたモデルを提案する．本稿ではこのモデルを Mixture of Domain Experts with a Domain Classifier (MoDE+DC) と呼ぶ．

図 3 の上図は MoDE+DC を示している．まず，入力文 $\boldsymbol{x} = (x_1, \ldots, x_T)$ がどの分野のテキストかを分野分類器 (Domain Classifier) が予測する．次に，その予測された分野のモデルを用いて入力文 $\boldsymbol{x}$ を解析する．

分野分類器として，任意の分類器を用いることができる．本研究では，素性ベクトル計算に Gated Recurrent Unit (GRU)[7] を用い，多クラス分類に softmax 関数を用いる．

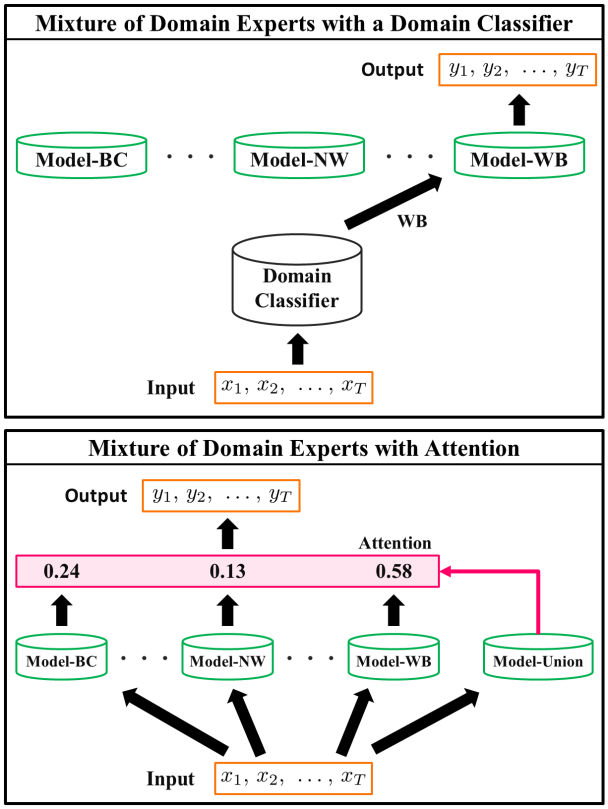

図 3 提案モデルの概略図.

具体的には，単方向の GRU を用いて隠れ状態 $\mathbf{h}_t$ を計算する．

$$\mathbf{h}_t = \mathrm{GRU}(\mathbf{x}_t, \mathbf{h}_{t-1}).$$

入力文 $\boldsymbol{x} = (x_1, \ldots x_T)$ の最後の単語 x_T に対して計算された隠れ状態 $\mathbf{h}_T$ を利用し，softmax 関数で各ラベルの確率分布を計算する．

$$\mathbf{d} = \mathrm{softmax}(\mathbf{W}^{dc}\, \mathbf{h}_T).$$

計算されたベクトル $\mathbf{d} \in \mathbb{R}^K$ は，次元が元分野数 K であり，各要素に各分野の予測確率を持つ．この要素の中で最大の確率値を持つ分野のインデックス $\hat{k}$ を最終的な予測として出力する．

$$\hat{k} = \underset{k}{\mathrm{argmax}}(\mathbf{d}[k]).$$

この予測結果にしたがい，インデックス $\hat{k}$ に該当する分野のエキスパートを用いて意味役割付与を行う．

$$Pr(\boldsymbol{y}|\boldsymbol{x}, p) = \mathrm{model}^{(\hat{k})}(\boldsymbol{x}, p)$$

各エキスパート $\mathrm{model}^{(k)}$ は，4.1 で述べた方法で構築したものを用いる．

4.3 複数エキスパート統合モデリング

複数エキスパート統合モデリングのインスタンス化として，各エキスパートの加重平均を利用するモデルを提案する．本稿ではこのモデルを Mixture of Domain Experts with Attention (MoDE+ATT) と呼ぶ．

図 3 の下図は MoDE+ATT を示している．まず，各分野のエキスパート $\mathrm{model}^{(k)}$ が入力文 $\boldsymbol{x} = (x_1 \ldots, x_T)$ と対象述語 p を受け取り，隠れ状態ベクトル $\mathbf{h}_T^{(k)}$ を計算する．次に，$\mathbf{h}_T^{(k)}$ を用いて各エキスパートに対する重みを計算し，加重平均をとる．最後に加重平均されたベクトルを用いて，ラベル系列の確率を求める．

定式的には，エキスパート k に対する重み q_k を以下のように求める．

$$q_k = [\mathbf{x}^{avg}\, ;\, \mathbf{h}_T^{(uni)}]^{\mathrm{T}}\, \mathbf{W}^{att}\, \mathbf{h}_T^{(k)}.$$

ここで，入力単語列の分散表現の平均ベクトル $\mathbf{x}^{avg}$ と UNION モデルの隠れ状態ベクトル $\mathbf{h}_T^{(uni)}$ を結合したベクトルを重み行列 $\mathbf{W}^{att}$ とかけ合わせ，そのあとに $\mathbf{h}_T^{(k)}$ をかける．計算された q_k を用いて，エキスパート k に対する重み a_k を以下のように求める．

$$a_k = \frac{\exp(q_k)}{\sum_{k'=1}^{K} \exp(q_{k'})}$$

もとめた重み a_k を各エキスパートの隠れ状態ベクトル $\mathbf{h}_t^{(k)}$ にかけて，以下のように足し合わせる．

$$\mathbf{h}_t = \sum_{k=1}^{K} a_k\, \mathbf{h}_t^{(k)}$$

もとめたベクトル $\{\mathbf{h}_t\}_1^T$ を入力とし， CRF を用いてラベル列 $\boldsymbol{y}$ の確率値を計算する．

$$Pr(\boldsymbol{y}\,|\boldsymbol{x}\; p) = \frac{1}{Z} \exp\Big(\sum_t \mathbf{W}^{tran}[y_{t-1}, y_t] + \mathbf{e}_t[y_t]\Big)$$

$$\mathbf{e}_t = \mathbf{W}^{emit}\, \mathbf{h}_t$$

ここで，$\mathbf{W}^{tran} \in \mathbb{R}^{|\mathcal{T}|\times|\mathcal{T}|}$ はラベルの遷移行列であり，$\mathbf{W}^{tran}[i,j]$ は $\mathbf{W}^{tran}$ の i 行 j 列目の要素を表す．また，各要素に生成スコアを持つベクトル $\mathbf{e}_t \in \mathbb{R}^{|\mathcal{T}|}$ は，重み行列 $\mathbf{W}^{emit} \in \mathbb{R}^{|\mathcal{T}|\times d_h}$ と加重平均されたベクトル $\mathbf{h}_t \in \mathbb{R}^{d_h}$ の行列積から計算される．Z は正規化項である．

4.4 学習

以下の目的関数を最小化することによってパラメータの学習を行う．

$$\mathcal{L}(\boldsymbol{\theta}) = -\sum_i \log Pr(\boldsymbol{y}_i|\boldsymbol{x}_i, p_i) + \frac{\lambda}{2}||\boldsymbol{\theta}||^2$$

ここで，係数 λ はハイパーパラメータである．MoDE+DC では，分野分類器のパラメータ $\mathbf{W}^{dc}$ のみを更新する．MoDE+ATT では，CRF のパラメータ $\{\mathbf{W}^{tran}, \mathbf{W}^{emit}\}$ と，重み q_k を計算する際のパラメータ $\mathbf{W}^{att}$ のみを更新する．両モデルとも，各エキスパートのパラメータは更新しない．

	学習		開発		評価	
	文数	述語数	文数	述語数	文数	述語数
BC	10,429	25,917	1,946	4,669	2,037	5,420
BN	9,723	29,225	1,172	3,626	1,252	3,797
MZ	6,911	24,165	642	2,158	780	2,656
NW	15,288	45,546	2,054	6,588	1,898	5,843
PT	15,263	34,977	1,075	2,594	1,217	2,831
TC	11,162	14,103	1,634	2,058	1,366	1,712
WB	6,411	14,971	1,080	2,217	929	2,200
ALL	75,187	188,904	9,603	23,910	9,479	24,459

表 1 データセットの統計.

5. 実験

5.1 実験設定

データセット

PropBank 形式スパン (句構造) 型意味役割付与のデータセットとして，CoNLL-2012 Shared Task[32] を用いる[*2]. このデータセットは以下の 7 つの分野から構成される.

- BC Broadcast Convesation
- BN Broadcast News
- MZ Magazine
- NW Newswire
- PT English Translation of the New Testament
- TC Telephone Conversation
- WB Weblogs and Newsgroups

表 1 に，このデータセットに含まれる文数・述語数を表す.

実装詳細

モデルの実装は，深層学習ライブラリ Theano[1] を利用した．エポック数は通常学習時は 100，Fine-Tuning 時は 30 に設定し，開発データの F1 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．単語埋め込みは SENNA[*3][9] を用い，学習時に Fine-Tuning は行わない．付録の表 A·1 に，モデルとその学習において使用するハイパーパラメータの詳細を示す.

5.2 既知分野適応実験

未知分野適応実験の前に，目標分野が既知の設定での適応実験を行う．この実験の目的は，未知分野適応のための性能基準を示すことである．未知分野適応は既知分野適応よりも困難であるため，既知分野適応実験の結果が未知分野適応実験における 1 つの目標値と見なすことができる.

[*2] 以下のページからダウンロード可能な OntoNotes のバージョンを用いる: http://cemantix.org/data/ontonotes.html. なお，使用する Document IDs は CoNLL-2012 で使用されるものと同一にそろえる.

[*3] http://ronan.collobert.com/senna/

	Target	Union	FineTuning
BC	72.10	79.02	**80.77**
BN	70.05	77.63	**79.92**
MZ	68.00	75.27	**77.08**
NW	71.59	75.05	**78.10**
PT	88.65	89.71	**91.60**
TC	75.60	80.46	**82.27**
WB	67.55	75.95	**78.83**
ALL	73.25	78.68	**80.97**

表 2 既知分野への適応における性能比較 (F1 値).

既知分野への適応実験の結果を表 2 に示す．表 2 の各行は，当該分野に対する F1 値を表す.「ALL」の行は全分野の F1 値を表す．各列は以下のモデルの F1 値を表す.

- Target: 目標分野と同一分野のデータのみから学習したモデル.
- Union: 全分野のデータから学習したモデル.
- FineTuning: Union モデルのパラメータを初期値として，各分野のデータのみを用いてモデルパラメータの再学習 (Fine-Tuning) をしたモデル.

表 2 の結果を見ると，Target が最も低い F1 値を示している．この理由として，各分野のデータのみを用いた場合，データ量が少ないため，よいモデルの構築に不十分であることが考えられる．また，FineTuning が Union よりも高い F1 値を表している．これは，目標分野のデータを用いてパラメータの再学習をすることによって，目標分野に特化したモデルが構築されるためであると考えられる．さらに，分野間で比較すると，PT に対する F1 値が顕著に高い．これは，平均文長が短く，単語異なり語数が少ないことに起因すると考えられる.

5.3 未知分野適応実験

未知分野適応における実験結果を表 3 に示す．表 3 の各行は，当該分野に対する F1 値を表す.「ALL」の行は全分野の F1 値を表す．各列は以下のモデルの F1 値を表す.

- Union: 全分野のデータから学習したモデル.
- Mode+dc: Mixture of Domain Experts with a Domain Classifier.
- Mode+att: Mixture of Domain Experts with Attention.

また，Mode+dc で使用する分野分類器の評価データに対する解析性能は，73.19%の正解率であった．以降の節で，Target-Covered/Zero-Shot 未知分野適応の結果をそれぞれ見ていく.

	Target-Covered DA			Zero-Shot DA		
	Union	Mode+dc	Mode+att	Union	Mode+dc	Mode+att
BC	79.02	80.26	**80.37**	72.26	72.80	**73.94**
BN	77.63	**79.42**	79.11	75.12	76.01	**76.46**
MZ	75.27	75.86	**76.62**	72.55	73.44	**74.50**
NW	75.05	77.71	**78.02**	69.01	69.84	**70.37**
PT	89.71	91.52	**91.67**	84.78	**85.82**	85.62
TC	80.46	**81.48**	81.47	75.88	77.41	**77.72**
WB	75.95	77.83	**77.85**	75.61	76.71	**77.02**
ALL	78.68	80.40	**80.55**	74.15	75.03	**75.60**

表 3　未知分野適応における性能比較 (F1 値).

Target-Covered 未知分野適応実験の結果

問題設定は 3.1 節で述べたように，**目標分野を学習データに含んでいる**．表 3 の結果を見ると，提案フレームワークの両モデルがベースラインの Union モデルを上回っている．この結果から，両モデルが Target-Covered 未知分野適応において有効であることがわかる．また，Mode+dc と Mode+att はほぼ同等の F1 値を記録している．

Zero-Shot 未知分野適応実験の結果

問題設定は 3.2 節で述べたように，**目標分野を学習データに含んでいない**．表 3 の結果を見ると，提案フレームワークの両モデルが Union モデルを上回っている．この結果から，両モデルが Zero-Shot 未知分野適応において有効であることがわかる．また，Mode+att が Mode+dc を上回る F1 値を記録しており，より有効に解析が行えていることがわかる．

6. 分析

5 章の実験結果から，各モデルの分野別の性能と提案モデルの有用性が確認された．本章では，それらの実験結果をより詳細に分析し，どのような部分に改善の余地があるかを明らかにする．

修正変換操作を用いた誤り分析

本節では，[20], [27] に従い，予測エラーを修正する変換操作を用いて，F1 値がどの程度改善するかを調査する．具体的には，以下の 4 つの修正変換規則を用いる．

- FixLabel　項のスパンが正解スパンと同一である場合，そのラベルを正解ラベルに修正する．
- FixSpan　項のラベルが正解のラベルと同一であり，かつ，そのスパンが正解スパンと重複している場合，そのスパンを正解スパンに修正する．
- DropArg　項のスパンがどの正解スパンとも重複していなければ，その項を削除する．
- AddArg　予想された他のどの項ともスパンが重複しない正解の項を追加する．

これらの変換規則を適用した結果を表 4 に示す．各数字は，それぞれの変換操作を適用した際に，F1 値が何ポイント改善したか (F1 改善値) を表している．全体的な傾向として，FixLabel 操作による F1 改善値が最も高く，意味役割ラベル付与に改善の余地が最もあると言える．次に FixSpan 操作による F1 改善値が高く，項のスパン同定にも改善の余地があることが示されている．対照的に，DropArg 操作による F1 改善値は最も低いため，余剰な項の予測は行われていないことがわかる．

それぞれの分野における結果を見ると，TC(Telephone Conversation) 分野では，FixSpan 操作の方が FixLabel 操作よりも F1 改善値が高い結果となっている．Zero-Shot 未知分野適応における BC(Broadcast Conversation) 分野でも，同様の傾向が見られる．これらの分野は，他の分野と異なり，会話文となっているため，項のスパンも他の分野と違いがあると考えられる．

意味役割ラベルに関する誤り分析

前節で，意味役割ラベル同定に関する改善の余地が大きいことが明らかになった．本節では，実際にどのようにラベルの予測が誤っているかを分析する．

図 4 は，頻出する意味役割ラベルの混同行列を表す．ラベル ARG0-ARG1 間の混同と ARG1-ARG2 間の混同が特に多いことがわかる．また，モデルが予測したラベル ARG2 が正解ラベル AM-DIR・AM-LOC・AM-MNR と多く混同されている．これらの混同は，各動詞フレームにおいて方向 (direction) や場所 (location) を意味する語句を「必須格 (Core-Argument)」とするか「周辺格 (Adjunct)」とするかの認定の難しさに起因すると考えられる．例えば，動詞フレーム *move.01* では distination や location を必須格 (ARG2) としているのに対し，*turn.01* では direction や location を周辺格 (AM-DIR や AM-LOC) としている．このような混同の傾向は，He ら [20] と同様の傾向である．

		Target-Covered DA				Zero-Shot DA			
		FixLabel	FixSpan	DropArg	AddArg	FixLabel	FixSpan	DropArg	AddArg
BC	Union	**6.23**	4.91	2.16	3.35	6.47	**9.09**	2.33	4.88
	Mode+dc	**5.75**	4.74	1.88	3.57	6.23	**8.15**	3.01	4.51
	Mode+att	**5.70**	4.61	1.86	3.45	6.03	**8.21**	2.33	4.55
BN	Union	**7.68**	5.48	1.98	2.78	**8.06**	5.34	1.76	4.25
	Mode+dc	**6.48**	5.24	1.84	2.95	**7.85**	5.70	2.24	3.38
	Mode+att	**6.70**	5.24	1.81	2.86	**7.72**	5.46	2.02	3.42
MZ	Union	**8.58**	5.67	2.27	3.49	**9.34**	7.30	1.79	4.59
	Mode+dc	**8.70**	5.43	1.92	3.92	**9.09**	6.39	2.54	3.57
	Mode+att	**8.61**	5.19	2.01	3.57	**8.76**	6.24	2.33	3.57
NW	Union	**7.38**	7.26	1.87	3.64	9.01	**9.38**	1.79	4.98
	Mode+dc	**6.71**	6.44	1.75	3.45	**8.96**	8.55	1.83	5.18
	Mode+att	**6.74**	6.09	1.77	3.35	**8.96**	8.35	1.79	4.91
PT	Union	**4.46**	2.40	0.79	0.92	**7.04**	3.28	0.95	1.59
	Mode+dc	**3.69**	1.78	0.59	0.89	**6.43**	3.09	0.82	1.65
	Mode+att	**3.59**	1.78	0.61	0.93	**6.51**	3.10	0.85	1.85
TC	Union	4.49	**5.26**	3.23	2.47	5.11	**6.36**	3.68	4.70
	Mode+dc	4.04	**4.62**	2.97	2.92	5.00	**5.59**	3.39	4.72
	Mode+att	4.06	**4.80**	3.00	3.28	4.64	**5.57**	3.45	4.78
WB	Union	**7.49**	6.80	1.84	3.63	**6.95**	6.68	1.72	4.66
	Mode+dc	**6.75**	6.26	1.64	3.56	**6.92**	5.96	1.95	4.28
	Mode+att	**6.81**	6.06	1.77	3.63	**6.87**	6.11	1.86	3.98
ALL	Union	**6.71**	5.52	1.97	2.98	**7.61**	7.25	1.94	4.30
	Mode+dc	**6.09**	5.05	1.75	3.07	**7.41**	6.66	2.23	4.01
	Mode+att	**6.15**	4.84	1.79	3.00	**7.29**	6.59	2.01	4.00

表 4 修正変換操作に基づく誤り分析．各数値は，各修正変換操作を施した場合の F1 値の改善ポイント数を表す．したがって，数値が大きいほど，改善の余地が大きいことを表す．

Target-Covered Unknown-DA

GOLD / PRED	A0			A1			A2			DIR			LOC			MNR			PNC			TMP		
	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT
A0		–		260	197	190	75	59	70	2	0	1	7	4	6	12	10	10	0	1	0	5	1	2
A1	189	202	187		–		292	239	256	13	8	8	32	26	17	54	48	45	14	13	14	31	15	20
A2	26	34	27	178	131	118		–		97	44	49	80	52	37	63	43	46	24	23	20	19	13	9
DIR	0	1	1	10	13	12	14	34	32		–		5	7	4	0	2	3	0	0	0	2	1	1
LOC	2	8	8	26	30	40	42	59	87	6	10	10		–		16	20	37	0	1	1	40	22	27
MNR	8	13	9	35	33	38	51	55	56	5	8	6	26	28	19		–		0	0	0	23	18	16
PNC	4	3	4	25	17	16	16	18	17	1	1	0	0	0	0	2	2	1		–		0	0	0
TMP	9	6	6	8	15	15	11	15	18	2	1	1	8	13	14	16	14	15	0	0	1		–	

Zero-Shot Unknown-DA

GOLD / PRED	A0			A1			A2			DIR			LOC			MNR			PNC			TMP		
	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT	Uni	DC	ATT
A0		–		289	261	252	80	77	73	1	3	3	5	6	5	13	14	10	0	0	0	5	4	3
A1	246	235	229		–		410	366	370	16	21	24	23	22	23	67	51	50	22	19	22	32	28	32
A2	40	34	30	210	197	188		–		63	59	64	49	59	53	68	66	62	35	26	29	15	8	10
DIR	0	0	0	22	13	12	31	37	37		–		8	7	8	7	4	3	0	0	0	1	3	1
LOC	6	4	2	33	39	32	64	57	58	12	9	6		–		27	21	20	1	0	0	27	17	16
MNR	9	8	8	35	38	33	69	78	79	3	4	5	16	24	24		–		1	1	1	26	25	26
PNC	3	1	1	23	16	13	23	15	18	0	0	0	0	0	0	2	1	1		–		1	0	0
TMP	7	8	6	20	14	17	24	24	22	2	2	2	23	21	20	25	24	21	2	1	1		–	

図 4 意味役割ラベル混同行列．

7. 関連研究

分野適応一般

これまで，分野適応に関する多くの研究がなされてきた [2], [3], [4], [6], [11], [12]．近年，ニューラルネットワークを用いた分野適応が盛んに研究されている．中でも，敵対的学習 (Adversarial Training)[15], [16], [38]，特徴量拡張 (Feature Augmentation)[25], [37]，自己学習 (Self-Training)[17], [33]，構造対応学習 (Structural Correspondence Learning)[44]，アンサンブル学習 (Ensemble Training)[14], [24]，重要度重み付け (Importance Weighting)[39] などのアプローチが分野適応実験で良い結果を報告している．言語処理の文脈では，Hal Daumé III[11] の特徴量拡張手法をニューラルネットワークに適応可能にした Kim ら [25] の手法が，意図分類 (Intent Classification) や Slot Filling で良い結果を報告している．

意味役割における分野適応

意味役割付与における分野適応では，Deep Belief Networks を用いて，目標分野のラベルなしデータを利用する手法が提案され，一定の成果を得ている [42]．また，WordNet や言語モデルを用いて特定の意味役割ラベルの精度向上を狙った手法 [13] や，単語の分散表現を使用しフレーム同定 (Frame Identification) の解析結果も報告されている [19]．これらの研究は依存構造意味役割付与 (Dependency-Based SRL)[13], [42] やフレームネット意味役割付与 (FrameNet SRL)[19] であるため，本研究で扱うスパン型意味役割付与 (Span-Based SRL) の結果と直接比較はできない．スパン型意味役割付与において，Huang ら [22] は隠れマルコフモデルを用いて教師なしでスパンの表現を学習することによって，分野外のテキストの解析精度向上を実現している．彼らの研究の問題設定は，単一の元分野から既知の目標分野に適応する手法であるため，複数元分野から未知の目標分野に適応する本研究の問題設定とは異なる．

未知分野適応

未知分野適応のアイディアは Blitzer ら [5] がすでに提案している．彼らの問題設定において，目標分野に関連する分野のデータを学習に用いることを前提としている点が，本研究の Target-Covered 未知分野適応と類似している．近年，Peng ら [31] は，画像処理の文脈において，Blitzer らの問題設定を Zero-Shot 未知分野適応に拡張している．本研究は，意味役割付与において Target-Covered/Zero-Shot 未知分野適応の問題を定式化し，包括的に調査した研究であると言える．

複数元分野適応

機械学習の文脈において，複数元分野適応 (Multi-Source Domain Adaptation) の理論的研究が進められ [10], [28]，言語処理の文脈でも，文書分類 [41] や機械翻訳 [8], [34] で主に研究が行われている．本研究に関連する手法として，Kim ら [24] は Mixture of Experts (MoE)[23], [36] に基づいた手法を提案しており，意図分類 (Intent Classification) や Slot Filling に適用している．しかし，彼らの手法は，目標分野が既知の場合にしか適用できない．したがって，本研究における手法は，未知分野にも適応可能なように彼らの手法を拡張したものとみなすことができる．

8. おわりに

本研究では，意味役割付与において，(1) 未知の分野への適応と (2) 複数元分野データを用いた適応の両方に取り組んだ．まず，未知分野適応を 2 つの異なる問題設定 (Target-Covered/Zero-Shot) に細分化し，定式化を行った．次に，それらを解くために，複数元分野データを利用した 2 つのモデリングフレームワークを提案した．各問題設定のもと，CoNLL-2012 Shared Task のデータセットを用いて，提案フレームワークからインスタンス化されたモデル (MoDE+DC，MoDE+ATT) の性能評価を行った．評価実験を通して，両モデルの有効性が明らかになると同時に，Zero-Shot 未知分野適応は Target-Covered 未知分野適応よりも困難であり，F1 値で 4-5 ポイント程度の差があることがわかった．2 つのモデルを比較すると，Target-Covered 未知分野ではほぼ同精度であったが，Zero-Shot 未知分野適応では MoDE+ATT が上回る結果となった．

さらに，解析結果に対して修正変換操作に基づいた誤り分析を行い，全体としては意味役割ラベル同定において改善の余地が大きいことが明らかになった．その中で，会話の分野では，項のスパン同定のほうがラベル同定よりも改善の余地が大きい傾向があった．また，意味役割ラベル混同行列の分析から，ラベル `ARG2` と `AM-DIR`・`AM-LOC` の混同が顕著に見られた．今後の課題として，これらの意味役割ラベル同定精度の改善や，教師ラベルなしデータを利用した分野適応手法の開発などが挙げられる．

謝辞 PFN の坪井祐太氏と，情報通信研究機構の藤田篤氏にさまざまなご教示を頂いたことを深謝する．

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] Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F. and Vaughan, J. W.: A theory of learning from different domains, *Machine learning*, Vol. 79, No. 1, pp. 151–175 (2010).

[3] Ben-David, S., Blitzer, J., Crammer, K. and Pereira, F.: Analysis of representations for domain adaptation, *Proceedings of NIPS*, pp. 137–144 (2007).

[4] Blitzer, J., Dredze, M., Pereira, F. et al.: Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification, *Proceedings of ACL*, pp. 440–447 (2007).

[5] Blitzer, J., Foster, D. P. and Kakade, S. M.: Zero-shot domain adaptation: A multi-view approach, *Technical Report TTI-TR-2009-1* (2009).

[6] Blitzer, J., McDonald, R. and Pereira, F.: Domain adaptation with structural correspondence learning, *Proceedings of EMNLP*, pp. 120–128 (2006).

[7] Cho, K., van Merrienboer, 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).

[8] Chu, C., Dabre, R. and Kurohashi, S.: An Empirical Comparison of Domain Adaptation Methods for Neural Machine Translation, *Proceedings of ACL*, pp. 385–391 (2017).

[9] 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).

[10] Crammer, K., Kearns, M. and Wortman, J.: Learning from multiple sources, *Journal of Machine Learning Research*, Vol. 9, No. Aug, pp. 1757–1774 (2008).

[11] Daumé III, H.: Frustratingly Easy Domain Adaptation, *Proceedings of ACL*, pp. 65–72 (2007).

[12] Daumé III, H. and Marcu, D.: Domain Adaptation for Statistical Classifiers, *Journal of Artificial Intelligence Research (JAIR)*, Vol. 26, pp. 101–126 (2006).

[13] 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 Transactions on Audio, Speech, and Language Processing*, Vol. 23, No. 11, pp. 1812–1823 (2015).

[14] French, G., Mackiewicz, M. and Fisher, M.: Self-ensembling for domain adaptation, *arXiv preprint arXiv:1706.05208* (2017).

[15] Ganin, Y. and Lempitsky, V.: Unsupervised domain adaptation by backpropagation, *Proceedings of ICML*, pp. 1180–1189 (2015).

[16] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M. and Lempitsky, V.: Domain-adversarial training of neural networks, *Journal of Machine Learning Research*, Vol. 17, No. 59, pp. 1–35 (2016).

[17] Golub, D., Huang, P.-S., He, X. and Deng, L.: Two-Stage Synthesis Networks for Transfer Learning in Machine Comprehension, *Proceedings of EMNLP*, pp. 846–855 (2017).

[18] 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).

[19] Hartmann, S., Kuznetsov, I., Martin, T. and Gurevych, I.: Out-of-domain FrameNet Semantic Role Labeling, *Proceedings of EACL*, pp. 471–482 (2017).

[20] He, L., Lee, K., Lewis, M. and Zettlemoyer, L.: Deep Semantic Role Labeling: What Works and What' s Next, *Proceedings of ACL*, pp. 473–483 (2017).

[21] Hovy, E., Marcus, M., Palmer, M., Ramshaw, L. and Weischedel, R.: OntoNotes: the 90% solution, *Proceedings of the human language technology conference of the NAACL, Companion Volume: Short Papers*, Association for Computational Linguistics, pp. 57–60 (2006).

[22] Huang, F. and Yates, A.: Open-Domain Semantic Role Labeling by Modeling Word Spans, *Proceedings of ACL*, pp. 968–978 (2010).

[23] Jacobs, R. A., Jordan, M. I., Nowlan, S. J. and Hinton, G. E.: Adaptive mixtures of local experts, *Neural computation*, Vol. 3, No. 1, pp. 79–87 (1991).

[24] Kim, Y.-B., Stratos, K. and Kim, D.: Domain Attention with an Ensemble of Experts, *Proceedings of ACL*, pp. 643–653 (2017).

[25] Kim, Y.-B., Stratos, K. and Sarikaya, R.: Frustratingly Easy Neural Domain Adaptation, *Proceedings of COLING*, pp. 387–396 (2016).

[26] Kingma, D. and Ba, J.: Adam: A Method for Stochastic Optimization, *arXiv preprint arXiv: 1412.6980* (2014).

[27] Kummerfeld, J. K., Hall, D., Curran, J. R. and Klein, D.: Parser Showdown at the Wall Street Corral: An Empirical Investigation of Error Types in Parser Output, *Proceedings of EMNLP*, pp. 1048–1059 (2012).

[28] Mansour, Y., Mohri, M. and Rostamizadeh, A.: Domain adaptation with multiple sources, *Proceedings of NIPS*, pp. 1041–1048 (2009).

[29] Marcheggiani, D., Frolov, A. and Titov, I.: A Simple and Accurate Syntax-Agnostic Neural Model for Dependency-based Semantic Role Labeling, *Proceedings of CoNLL*, pp. 411–420 (2017).

[30] Miller, G. A.: WordNet: a lexical database for English, *Communications of the ACM*, Vol. 38, No. 11, pp. 39–41 (1995).

[31] Peng, K.-C., Wu, Z. and Ernst, J.: Zero-Shot Deep Domain Adaptation, *arXiv preprint 1707.01922* (2017).

[32] Pradhan, S., Moschitti, A., Xue, N., Uryupina, O. and Zhang, Y.: CoNLL-2012 Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes, *Proceedings of EMNLP-CoNLL*, pp. 1–40 (2012).

[33] Saito, K., Ushiku, Y. and Harada, T.: Asymmetric Tri-training for Unsupervised Domain Adaptation, *Proceedings of ICML*, pp. 2988–2997 (2017).

[34] Sajjad, H., Durrani, N., Dalvi, F., Belinkov, Y. and Vogel, S.: Neural Machine Translation Training in a Multi-Domain Scenario, *arXiv preprint arXiv:1708.08712* (2017).

[35] Schuster, M. and Paliwal, K. K.: Bidirectional recurrent neural networks, *IEEE Transactions on Signal Processing*, pp. 2673–2681 (1997).

[36] Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G. and Dean, J.: Outrageously large neural networks: The sparsely-gated mixture-of-experts layer, *Proceedings of ICLR* (2017).

[37] Sun, B., Feng, J. and Saenko, K.: Return of Frustratingly Easy Domain Adaptation, *Proceedings of AAAI*, pp. 2058–2065 (2016).

[38] Tzeng, E., Hoffman, J., Saenko, K. and Darrell, T.: Adversarial discriminative domain adaptation, *arXiv preprint arXiv:1312.6026* (2017).
[39] Wang, R., Utiyama, M., Liu, L., Chen, K. and Sumita, E.: Instance Weighting for Neural Machine Translation Domain Adaptation, *Proceedings of EMNLP*, pp. 1483–1489 (2017).
[40] Weischedel, R., Pradhan, S., Ramshaw, L., Kaufman, J., Franchini, M., El-Bachouti, M., Xue, N., Palmer, M., Hwang, J. D., Bonial, C. et al.: OntoNotes Release 5.0 (2012).
[41] Wu, F. and Huang, Y.: Sentiment Domain Adaptation with Multiple Sources, *Proceedings of ACL*, pp. 301–310 (2016).
[42] Yang, H., Zhuang, T. and Zong, C.: Domain Adaptation for Syntactic and Semantic Dependency Parsing Using Deep Belief Networks, *Transactions of ACL*, Vol. 3, pp. 271–282 (2015).
[43] Zhou, J. and Xu, W.: End-to-end learning of semantic role labeling using recurrent neural networks, *Proceedings of ACL-IJCNLP*, pp. 1127–1137 (2015).
[44] Ziser, Y. and Reichart, R.: Neural Structural Correspondence Learning for Domain Adaptation, *Proceedings of ACL*, pp. 400–410 (2017).

付　　録

A.1 基本モデルの詳細

基本モデルとして，最先端の意味役割付与モデル (Deep Recurrent Model; DRM)[20], [43] を用いる．DRM は，入力として単語列 $\boldsymbol{x} = (x_1, \ldots, x_T)$ と解析対象の述語 p を受け取り，ラベル列 $\boldsymbol{y} = (y_1, \ldots, y_T)$ の確率値を返す．

$$Pr(\boldsymbol{y}|\boldsymbol{x}, p) = f(\boldsymbol{x}, p).$$

DRM 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: CRF を用い，ラベル列の確率値を求める．

入力層 (Input Layer) では，入力文の各単語 $x_1, \ldots, x_T$ に素性ベクトル $\mathbf{h}_1^{(0)}, \ldots \mathbf{h}_T^{(0)}$ を割り当てる．各素性ベクトル $\mathbf{h}_t^{(0)}$ は以下のように計算される．

$$\mathbf{h}_t^{(0)} = \mathbf{x}_t^{word} \oplus \mathbf{x}_t^{mark}$$

ここで，$\mathbf{x}_t^{word} \in \mathbb{R}^{d_{word} \times |\mathcal{V}|}$ とは単語 x_t の分散表現を表し，$\mathbf{x}_t^{mark} \in \mathbb{R}^{d_{mark} \times 2}$ は単語 x_t が解析対象である述語 p か否かの 2 値インデックスの分散表現を表している．これら 2 種類のベクトル表現を結合 ($\oplus$) し，ベクトル $\mathbf{h}_t^{(0)} \in \mathbb{R}^{d_{word}+d_{mark}}$ が得られる．

中間層では，入力層で計算された素性ベクトル $\mathbf{h}_t^{(0)}$ を，深層双方向型リカレントニューラルネットワーク (Bi-RNNs) に入力として与える．特に，奇数番目の層では系列を左から右に，偶数番目の層では右から左に処理するような，波形に多層化した Bi-RNNs(Interleaving Bi-RNNs) を用いる．

$$\mathbf{h}_t^{(\ell)} = \begin{cases} g^{(\ell)}(\mathbf{h}_t^{(\ell-1)}, \mathbf{h}_{t-1}^{(\ell)}) & (\ell = \text{odd}) \\ g^{(\ell)}(\mathbf{h}_t^{(\ell-1)}, \mathbf{h}_{t+1}^{(\ell)}) & (\ell = \text{even}) \end{cases}$$

ここで，奇数番目の層では，各時刻 t において，$\ell - 1$ 層目の RNN の素性ベクトル $\mathbf{h}_t^{(\ell-1)}$ と時刻 $t-1$ での素性ベクトル $\mathbf{h}_{t-1}^{(\ell)}$ を入力とし，素性ベクトル $\mathbf{h}_t^{(\ell)}$ を計算する．同様に，偶数番目の層では，右から左に伝搬するため，$\mathbf{h}_{t-1}^{(\ell)}$ の代わりに $\mathbf{h}_{t+1}^{(\ell)}$ が入力に用いられる．本研究では，関数 $g(\cdot)$ として Gated Recurrent Unit(GRU)[7] を用いる．

出力層では，一次の条件付確率場 (Linear-Chain Conditional Random Fields; Linear-Chain CRF) を用いて，ラベル列 $\boldsymbol{y}$ の確率値を計算する．

$$Pr(\boldsymbol{y} \mid \boldsymbol{x}, p) = \frac{1}{Z} \exp(\sum_t \mathbf{W}^{tran}[y_{t-1}, y_t] + \mathbf{e}_t[y_t])$$

$$\mathbf{e}_t = \mathbf{W}^{emit}\, \mathbf{h}_t^{(L)}$$

ここで，$\mathbf{W}^{tran} \in \mathbb{R}^{|\mathcal{T}| \times |\mathcal{T}|}$ はラベルの遷移行列であり，$\mathbf{W}^{tran}[i, j]$ は $\mathbf{W}^{tran}$ の i 行 j 列目の要素を表す．$\mathcal{T}$ は可能なラベル集合であり，$y_t \in \mathcal{T}$ である．また，各要素に生成スコアを持つベクトル $\mathbf{e}_t \in \mathbb{R}^{|\mathcal{T}|}$ は，重み行列 $\mathbf{W}^{emit} \in \mathbb{R}^{|\mathcal{T}| \times d_h}$ と Bi-RNNs の最終 L 層目の素性ベクトル $\mathbf{h}_t^{(L)} \in \mathbb{R}^{d_h}$ の行列積である．Z は正規化項である．

A.2 実験に用いたハイパーパラメータ

パラメータ名	値
単語埋め込み次元	50
隠れ層次元	128
隠れ層数	4
ミニバッチサイズ	32
最適化アルゴリズム	Adam[26]
学習率	0.001(通常)，0.0001(Fine-Tuning)
L2 正則化係数	{ 0.0001, 0.0005, 0.001 }

表 A·1 実験に使用するハイパーパラメータ．

5 章で用いたハイパーパラメータを表 A·1 に示す．なお，Adam のハイパーパラメータ β_1 と β_2 は文献 [26] で推奨されている 0.9 と 0.999 にそれぞれ設定している．