注意型ニューラルネットワークによる非明示的談話関係解析の精度向上

寺田 凛太郎† 柴田 知秀†‡ 黒橋 禎夫†‡
†京都大学 ‡科学技術振興機構 CREST
†‡{terada, shibata, kuro}@nlp.ist.i.kyoto-u.ac.jp

1 はじめに

談話関係解析とは文章中の談話単位間の意味的関係 (例: Expansion や Contrast など) を同定するタスクである. 質問応答や対話システムなどのアプリケーションの高度化のために談話関係解析の精度向上は重要であり, CoNLL2015, 2016 での Shared Task にも指定されている [13][12].

談話関係タグ付きコーパス Penn Discourse Treebank (PDTB)[8] では二つの談話単位間の関係, およびそれが明示的であるか (関係を明示する接続詞があるか) どうかがタグ付けしてある. PDTB に含まれるデータの例を以下に示す. イタリック体になっているのが 1 つめの談話単位 (Argument1), 太字になっているのが 2 つ目の談話単位 (Argument2) である.

> *The brokerage firms learned a lesson the last time around*, when frightened investors flooded the phone lines and fled the market in a panic. **This time, the firms were ready**.
> 談話関係: Comparison.Contrast

談話関係解析は, 以下の 3 つのサブタスクに分解することができる.

1. (明示的な関係に対して) 接続詞の同定
2. 関係を持つ 2 つの談話単位の同定
3. 2 つの談話単位間の関係の同定

これらのサブタスクそれぞれについて, CoNLL2016 Shared Task における最善のモデルの F1-score を表 1 に示す. 談話単位の同定については, 正解データと 7 割以上のトークンがマッチすれば正解とする部分マッチのスコアを載せている. この表を見ると他のサブタスクに比べて, 非明示的な関係の同定が非常に難しいことが分かる. 非明示的な関係を解析するにあたって, 上に示した例では, 同一の主語をもつ述語であり, 対比的な意味を持つ *learned a lesson* と *were ready* が大きな手がかりとなる. しかし, CoNLL2016 Shared Task で最高のスコアを達成したニューラルネットモデルでも, どの語に注目するべきかを明示的に考慮する仕組みは取り入れられていない.

表 1: CoNLL2016 shared task における各サブタスクの最高 F1-score

サブタスク	明示的関係	非明示的関係
接続詞の同定	98.92	-
談話単位の同定	74.89	86.88
関係の同定	90.13	40.91

そこで本研究では, ニューラルネットモデルにアテンション機構を取り入れることで注目すべき語を予測し非明示的談話関係の分類精度を向上させる手法を提案する. Word Embedding と LSTM の出力を利用するベースラインモデルにアテンション機構を取り入れることで, PDTB を用いた実験において F 値を約 9 ポイント向上させた.

2 関連研究

PDTB に対する非明示的な談話関係判別の手法としては, ごく最近までは人手で設計した大量の特徴量を分類器の入力とするのが主であった [3] [7]. しかし, 近年は CoNLL2016 shared task での最高精度を達した畳み込みニューラルネットを利用したモデル [11] をはじめ, Word Embedding の和と積を用いたモデル [9] やマルチタスクにすることで正則化したモデル [5] などが従来の人手で設計した特徴量ベースのモデルよりも高い精度をあげている.

文章の感情分析のタスクなどでも文の構造に応じて再帰的にベクトルを構成する再帰型ニューラルネット [10] や, 畳み込みニューラルネット [14] が特徴量ベー

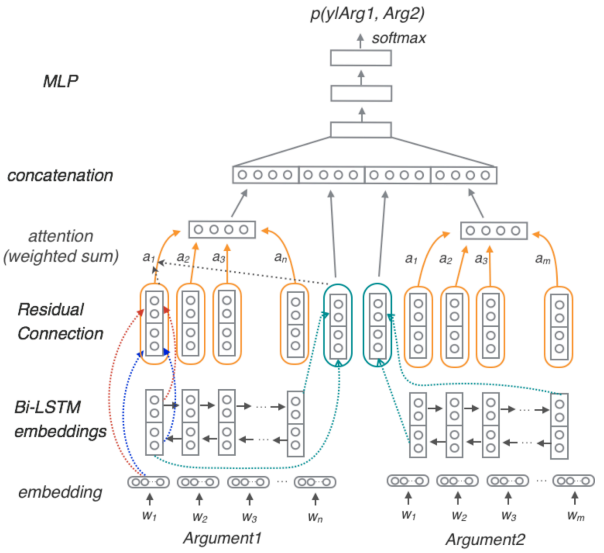


図 1: モデル図

スの手法よりも高い性能をあげている.

3 手法

本節では, 非明示的な談話関係を推定するニューラルネットモデルについて説明する.

3.1 ベースラインモデル

単語の分散表現を入力とした双方向 LSTM の各単語時点での隠れ層を用いることでより文脈の情報を考慮した単語の分散表現が得られることが知られている. この双方向 LSTM を用いて $Arg1, Arg2$ 内の単語の分散表現を得る.

まず, $Arg1$ に含まれる j 番目の単語 w_j^1 の d 次元の単語の分散表現を $\mathbf{x}_j^1 \in \mathbb{R}^d$ と表わすこととする. これによりそれぞれ n 単語, m 単語を含む $Arg1, Arg2$ はそれぞれ以下のように表わされる.

$$Arg1 : [\mathbf{x}_1^1, \mathbf{x}_2^1, \cdots, \mathbf{x}_n^1]$$
$$Arg2 : [\mathbf{x}_1^2, \mathbf{x}_2^2, \cdots, \mathbf{x}_m^2]$$

双方向 LSTM を用いた単語の分散表現では, $Arg1$, $Arg2$ の j 番目の単語の分散表現 $\mathbf{h}_j$ は, 単語を文の先頭から順に読んだ LSTM と文の最後から順に読んだ 2 つの LSTM の j 単語目の時点での隠れ層 $\overrightarrow{\mathbf{h}}_j$ と $\overleftarrow{\mathbf{h}}_j$ を用いてそれぞれ以下のように表わされる.

$$\mathbf{h}_j^1 = [\overrightarrow{\mathbf{h}}_j^1; \overleftarrow{\mathbf{h}}_j^1] \quad (1)$$
$$\mathbf{h}_j^2 = [\overrightarrow{\mathbf{h}}_j^2; \overleftarrow{\mathbf{h}}_j^2] \quad (2)$$

単語の分散表現 $\mathbf{h}_j$ を用いて, $Arg1$ と $Arg2$ の分散表現 $\mathbf{R}^1, \mathbf{R}^2$ を求める方法として以下のようにそれぞれに含まれる単語の表現の平均をとることが考えられる.

$$\mathbf{R}_{average}^1 = \frac{1}{n}\sum_{j=1}^{n}\mathbf{h}_j^1 \quad (3)$$
$$\mathbf{R}_{average}^2 = \frac{1}{m}\sum_{j=1}^{m}\mathbf{h}_j^2 \quad (4)$$

また, 他には順方向の LSTM の最終的な隠れ層と, 逆方向の LSTM の最終的な隠れ層を連結する以下のような方法も考えられる.

$$\mathbf{R}_{lstm}^1 = [\overrightarrow{\mathbf{h}}_n^1; \overleftarrow{\mathbf{h}}_1^1] \quad (5)$$
$$\mathbf{R}_{lstm}^2 = [\overrightarrow{\mathbf{h}}_m^2; \overleftarrow{\mathbf{h}}_1^2] \quad (6)$$

ベースラインモデルとして, これらを全て結合した $\mathbf{h}_{in}$ を多層パーセプトロン (MLP) に入力し, ソフトマックスをかけることで最終的な出力である各ラベルの確率分布を得るモデルを採用した.

$$p(\mathbf{y}|Arg1, Arg2) = softmax(MLP(\mathbf{h}_{in})) \quad (7)$$

3.2 アテンション機構を用いた重み付け

前節の方法では全ての単語に等しく重みを付けているが, 本来は非明示的談話関係の分類に有用な単語に対してより大きな重み付けをしたい. そのためにアテンション機構と呼ばれる仕組みを利用して各単語を足し合わせる際の重み係数を決定する.

$Arg1$ 内の各単語を足す際の重みを決めるにあたり, $Arg2$ の分散表現を用いて以下の式のように j 番目の単語の重み a_j^1 を決定する. $Arg2$ の分散表現として $\mathbf{R}_{lstm}^2 = [\overrightarrow{\mathbf{h}}_m^2; \overleftarrow{\mathbf{h}}_1^2]$ を用いる.

$$\mathbf{s}_j^1 = \tanh\left(\mathbf{W}_h\mathbf{x}_j^1 + \mathbf{W}_R\mathbf{R}_{lstm}^2\right) \quad (8)$$
$$\widetilde{a}_j^1 = \mathbf{W}_a\mathbf{s}_j^1 \quad (9)$$
$$a_j^1 = \frac{\exp\left(\widetilde{a}_j^1\right)}{\sum_{k=1}^{n}\exp\left(\widetilde{a}_k^1\right)} \quad (10)$$

ここで, $\mathbf{W}_h, \mathbf{W}_a, \mathbf{W}_R$ は学習するパラメータである. $Arg2$ の各単語の重みについても同様にして決める.

最終的には, 以下の式のようにアテンションによって重み付けした単語の和, および文の分散表現を $Arg1$,

Arg2 それぞれについて作成し, それらを結合する.

$$\mathbf{h}^1_{att} = \sum_{j=1}^{n} a^1_j \mathbf{h}^1_j \quad (11)$$

$$\mathbf{h}^2_{att} = \sum_{j=1}^{m} a^2_j \mathbf{h}^2_j \quad (12)$$

$$\mathbf{h}_{attin} = [\mathbf{h}^1_{att}; \mathbf{R}^1_{lstm}; \mathbf{h}^2_{att}; \mathbf{R}^2_{lstm}] \quad (13)$$

3.3 Residual Connection

Residual Connection[2] とは, ニューラルネットのある層においてその出力に入力を足すことで, 変換が不要な層における恒等写像の学習を容易にし, また勾配の伝播経路が短くなることで多層のネットワークの学習を容易にする接続である. Residual Connection を導入することによって双方向 LSTM を多層にし, より複雑な文脈の情報を考慮することができる. 入力と N 層双方向 LSTM の出力の間に Residual Connection を導入した場合の j 番目の単語の分散表現 $\mathbf{x}^{res}_j$ は以下の式のように表せる.

$$\overrightarrow{\mathbf{h}}^i_j = \overrightarrow{LSTM}^i(\overrightarrow{\mathbf{h}}^{i-1}_j, \overrightarrow{\mathbf{h}}^i_{j-1}) \quad (14)$$

$$\overleftarrow{\mathbf{h}}^i_j = \overleftarrow{LSTM}^i(\overleftarrow{\mathbf{h}}^{i-1}_j, \overleftarrow{\mathbf{h}}^i_{j+1}) \quad (15)$$

$$\mathbf{x}^{res}_j = [\mathbf{x}_j + \overrightarrow{\mathbf{h}}^N_j; \mathbf{x}_j + \overleftarrow{\mathbf{h}}^N_j] \quad (16)$$

ここで, $LSTM^i$ は i 層目の LSTM を, $\mathbf{h}^i_j$ は i 層目の LSTM における j 単語目時点での隠れ層を表している.

最終的に, 図 1 のようなモデルを用いて, 談話関係の予測を行う.

4 実験

4.1 データ

データセットとして, 24 個のセクションに区切られている PDTB のうち, セクション 2-20 を訓練セット, 0-1 を評価セット, 21-22 をテストセットとして用いた. PDTB ではラベルが階層化されており, どの階層まで細分化したラベルを用いるかなどは研究により差異がある. 本稿では Liu2016[4] などにならい, PDTB で定められている一番上の階層の 4 つのラベルを使用することとした. 表 2 にラベルの分布を示す.

4.2 実験設定

ハイパーパラメータを表 3 に示す. 各単語の Word Embedding は, continuous bag-of-words[6] を用いて Google News Corpus で事前に訓練した値で, 他のユニットに関してはバイアス項は 0, それ以外は (-0.10, 0.10) の一様分布からランダムにサンプリングした値で初期化した.

表 2: PDTB 内のデータのラベル分布

label	Train	Dev	Test
Comparison	1,855	189	145
Contingency	3,236	281	273
Expansion	6,673	638	538
Temporal	582	48	55
Total	12,346	1,156	1,011

表 3: ハイパーパラメータ

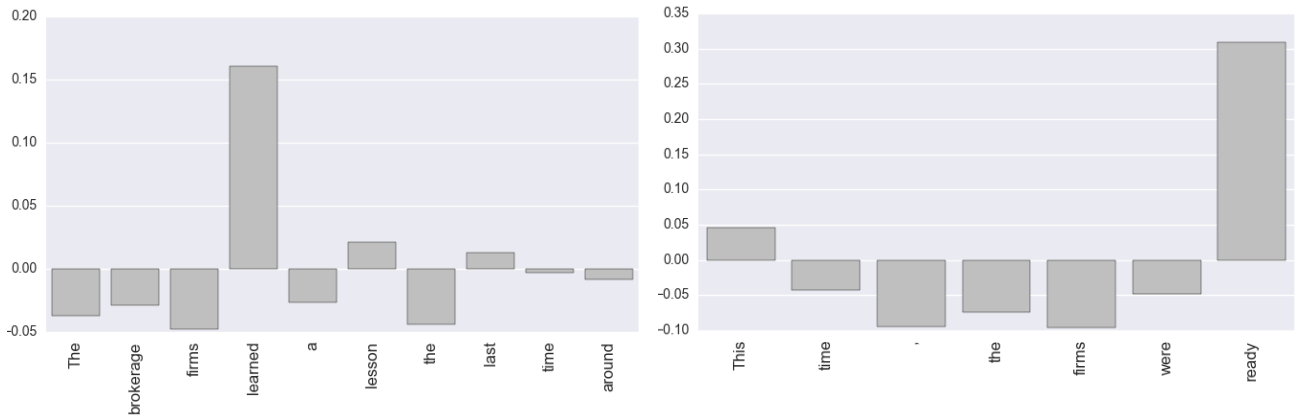
model	
# units of embedding layer	300
# units of lstm hidden state	300
# units of hidden layers	150
# hidden layers of MLP	2
activation function of MLP	tanh
Dropout rate (embedding)	0.2
Dropout rate (MLP)	0.5
optimizer	
learning rate	0.0001
alpha	0.95
momentum	0.9

モデルの訓練は全てモデルの出力と正解データの交差エントロピー誤差を誤差逆伝播法を用いて最適化することで行った. 最適化手法には Graves の提案した RMSProp[1] を用いた. 100epoch 訓練を行った中で, 評価セットでのスコアが最も高かったところで訓練を止めた状態でテストセットでテストを行った.

5 結果

各モデルの F1-score を表 4 に示す. 単純な平均をとるベースラインモデルに対して, アテンション機構を組み込むことで F 値が 7.2 ポイント上昇した. さらに, 単純に双方向 LSTM を多層化するだけではむしろスコアが落ちる結果となったが, Residual Connection を導入することで F 値が 1.7 ポイント上昇した.

3 層双方向 LSTM に Residual Connection を追加したモデルで得られたアテンションの重みを図 4 に示す. 図では平均を取った場合の重みとの差を縦軸にしている. これを見ると, *learend a leeson* と *ready* といった対比の関係にある 2 語に注目することができているといえる.



Arg1: The brokerage firms learned a lesson the last time around

Arg2: This time , the firms were ready

図 4: アテンションによる重み係数と平均を取る場合の重み係数 (1/単語数) との差

表 4: 各モデルの F1-score

手法	スコア
Liu2016	46.3
ベースライン	46.9
＋ アテンション	54.1
＋ 双方向 LSTM 多層化 (3 層)	51.9
＋ Residual Connection	54.4
＋ 双方向 LSTM 多層化 (3 層)	**55.8**

6 まとめと今後の課題

本研究では, 非明示的な談話関係解析における有効なアテンションの取り方, 及び Residual Connection を導入した双方向 LSTM の有効性を示し, ベースライン手法に対して F 値を 9 ポイント向上させた. 今後の予定としては, 異なるラベル, または異なるデータセットで本手法の有効性を示すことや, テキスト含意認識などの他のタスクへ適用することなどがあげられる.

参考文献

[1] Alex Graves. Generating sequences with recurrent neural networks. arXiv:1308.0850, 2013.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. arXiv:1512.03385, 2015.

[3] Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. A PDTB-styled end-to-end discourse parser. *Natural Language Engineering*, pp. 1–34, 2014.

[4] Yang Liu and Sujian Li. Recognizing implicit discourse relations via repeated reading: Neural networks with multi-level attention. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1224–1233, Austin, Texas, November 2016. Association for Computational Linguistics.

[5] Yang Liu, Sujian Li, Xiaodong Zhang, and Zhifang Sui. Implicit discourse relation classification via multi-task neural networks, 2016.

[6] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. arXiv:1301.3781, 2013.

[7] Emily Pitler, Annie Louis, and Ani Nenkova. Automatic sense prediction for implicit discourse relations in text. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pp. 683–691, Suntec, Singapore, August 2009. Association for Computational Linguistics.

[8] Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. The penn discourse treebank 2.0. In *Proceedings of LREC*, 2008.

[9] Niko Schenk, Christian Chiarcos, Kathrin Donandt, Samuel Rönnqvist, Evgeny Stepanov, and Giuseppe Riccardi. Do we really need all those rich linguistic features? a neural network-based approach to implicit sense labeling. In *Proceedings of the CoNLL-16 shared task*, pp. 41–49. Association for Computational Linguistics, 2016.

[10] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on EMNLP*, pp. 1631–1642, Seattle, Washington, USA, 2013.

[11] Jianxiang Wang and Man Lan. Two end-to-end shallow discourse parsers for English and Chinese in conll-2016 shared task. In *Proceedings of the CoNLL-16 shared task*, pp. 33–40. Association for Computational Linguistics, 2016.

[12] Nianwen Xue. Proceedings of the conll-16 shared task. In *Proceedings of the CoNLL-16 shared task*. Association for Computational Linguistics, 2016.

[13] Nianwen Xue, Tou Hwee Ng, Sameer Pradhan, Rashmi Prasad, Christopher Bryant, and Attapol Rutherford. The conll-2015 shared task on shallow discourse parsing. In *Proceedings of the CoNLL-2015 Shared Task*, pp. 1–16. Association for Computational Linguistics, 2015.

[14] Kim Yoon. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on EMNLP*, pp. 1746–1751, Doha, Qatar, 2014.