

階層構造を考慮したトップダウン談話構造解析

小林 尚輝¹ 平尾 努² 上垣外 英剛¹ 奥村 学¹ 永田 昌明²

¹ 東京工業大学 ² NTT コミュニケーション科学基礎研究所

1 はじめに

文を構成する単語の並びが構文構造を構成するように、最小の談話単位として EDU (Elementary Discourse Unit) を仮定すると、文書を構成する EDU の並びは談話構造を構成する。談話構造は文書要約、質問応答、意見分析など、文書を対象とした言語処理技術において有用であることから、近年重要視されている。

談話構造を表現する理論としては、本研究でも用いる修辞構造理論 (RST; Rhetorical Structure Theory) [5] が代表的なものの1つである。修辞構造理論では、原因/結果、例示、詳細化など、約 20 の修辞関係の多くを、核 (nucleus) となる談話単位に対し、衛星 (satellite) となる談話単位が修飾するもの¹として定義し、それらの談話単位間に関係が成立することで、より大きな談話単位を構成するもの²と考える。したがって、修辞構造理論を元にした談話構造は、EDU を葉ノードとした二分木として表現される。また、談話構造解析は、スパン (EDU そのものあるいは複数の EDU の系列) の解析 (解析木の構築)、2 つのスパンの核性ラベルの推定、スパン間の関係ラベルの推定の 3 つのタスクに分割される。

従来の談話構造解析手法の多くは、葉ノードである EDU から開始し、それらをボトムアップに組み上げていくことで、談話構造木を構築している [4]。しかし、ボトムアップな解析手法は動的計画法を用いた場合に通常、入力系列の長さの三乗の計算量を必要とする。また、計算時間削減のために枝刈りを併用した場合には、出力結果が葉ノード近辺の解析結果に依存しやすくなっていくという傾向がある。この傾向は、段落間関係のような、応用タスクにおいて利用価値が高い情報が根ノード近辺に存在していることから望ましくない。また、必ずしもボトムアップに木を組み上げない shift-reduce 談話構造解析手法 [10, 11] においても、通常一方向でのみ解析木を構築することから、葉ノード近辺の解析結果に影響を受けやすいという点では変わらない。

上述の問題を踏まえ、本研究では、トップダウンに談話構造解析を行う手法を提案する。また、通常の談話構造解析では、葉ノードが EDU である単一の木として談話構造をとらえるが、本研究では、文書は段落を葉ノードとした木、段落は文を葉ノードとした木、文は EDU を葉ノードとした木として独立に扱うことで、探索空間を階層的に分割して解析を行う。

RST Discourse Treebank (RST-DTB) [2] を用いた実験では、提案手法はスパンの評価において現在最高

¹対比、並列など、一部の関係は、2 つの核となる談話単位間に成り立つ。

精度の談話構造解析器と同等の精度を達成した。また、階層構造を考慮することにより、スパン、核、関係のいずれの評価においても解析精度が改善されることを示した。

2 関連研究

談話構造木は隣接するスパンを結合してそれよりも大きなスパンを構成するという手続きをボトムアップに繰り返すことで得られる。これは、句構造解析と同じであるが、文ではなく文書を対象とした解析であるため、入力系列中の単位の数、つまり EDU の数が非常に多い。このため、解析アルゴリズムの中でも比較的計算量の少ない shift-reduce 法がよく用いられる。従来型の素性に基づく shift-reduce 談話構造解析手法としてはスパンの解析、核性ラベル推定と関係ラベル推定を二段階に分割して解析する手法 [10] が最高精度を達成している。また、ニューラルネットワークを用いた shift-reduce 談話構造解析手法としては、LSTM に依存構造木の情報を入力し、構造学習を行う手法 [11] が最高精度を達成している。

しかし、これらの手法は葉ノード近辺のスパンの解析結果の影響を受けやすく、また、EDU を最小単位として扱うために探索空間が増大してしまうという問題がある。特に後段の処理での談話構造解析結果の利用を考える場合には、多くのタスクで現在あまり考慮されていない文間や段落間の関係が重要であり、根ノードに近い構造が重視される手法が有用であると考えられる。高精度な談話構造解析手法の中には、動的計画法に基づき、ボトムアップに解析を行う [4] ことで、このような問題を回避できる手法も存在するが、計算量の増大に比して解析精度の向上は大きくない。

3 提案手法

EDU の系列を入力として談話構造木の構築を行う従来手法とは異なり、提案手法では入力を段落、文、EDU を最小単位とする階層に分割し、各階層に位置する談話構造木の部分木を独立に構築する。スパン間の関係ラベルについては解析木の構築と核性ラベルの推定を行った後に独立して推定する。以下の節では、それぞれの手法の詳細について説明する。

3.1 階層に基づく解析単位の分割

提案手法において、談話構造木は階層的に分割された次の三つの解析単位における部分木により構成さ

Algorithm 1 トップダウン談話構造解析

```
1: function PARSE ( $i, j$ )
2:   if  $i - j = 1$  then
3:      $\triangleright$  葉ノードに到達したため解析を終了
4:     return 0
5:   else
6:      $\triangleright$  分割個所を推定
7:      $\hat{k} = SPLIT(i, j)$ 
8:      $\triangleright$  スパンのラベルを推定
9:      $\hat{l} = LABEL(i, j, \hat{k})$ 
10:     $\triangleright$  再帰的に解析を実行
11:     $left\_loss = PARSE(i, \hat{k})$ 
12:     $right\_loss = PARSE(\hat{k}, j)$ 
13:    return  $split\_loss + label\_loss + left\_loss +$ 
14:       $right\_loss$ 
15:   end if
16: end function
```

れる。

- 文書全体をスパンとしてカバーし、各段落を葉ノードとする木
- 対象の段落全体をスパンとしてカバーし、各文を葉ノードとする木
- 対象の文全体をスパンとしてカバーし、各 EDU を葉ノードとする木

各部分木はトップダウン談話構造解析器を用いて独立に解析され、それぞれの解析単位における部分木を結合することで文書全体を対象とする談話構造木が構築される。

3.2 各階層におけるトップダウン談話構造解析

階層的に分割された談話構造木の各階層における部分木は根ノードから葉ノードの方向にトップダウンで構築される。提案手法では、句構造解析におけるトップダウン構文解析手法の一つである Minimal Span-Based Neural Constituency Parser[9]と同様の方法で解析木を構築している。提案手法のアルゴリズムの概略をアルゴリズム 1 に示す。提案手法では、入力された葉ノードの系列を再帰的に二つのスパンに分割し、それぞれについて核か衛星のどちらのラベルであるかを推定するという手順を、分割されたスパンが葉ノードとなるまで繰り返す。以下の節ではスパンの表現、スパンの分割、核性ラベルの推定、使用する分散表現について詳細を説明する。

3.2.1 スパンのベクトル表現

3.2.4 節で述べる形式で表現される入力の葉ノードの系列は両方向の LSTM に入力され、これを元に、ス

パンのベクトル表現を得る。スパン (i, j) のベクトル表現は、スパン境界の順方向 LSTM の隠れ状態 f_i, f_j と逆方向 LSTM の隠れ状態 b_i, b_j の差分により、ベクトルの結合を意味する演算記号 $[\cdot]$ を用いて以下の式のように表現される。

$$s_{ij} = [f_j - f_i; b_i - b_j] \quad (1)$$

3.2.2 スパンの分割

スパン (i, j) の分割点 \hat{k} はスコア関数 $s_{split}(i, j, k)$ を用いて、 $i < k < j$ を満たす分割候補から

$$\hat{k} = SPLIT(i, j) = \underset{k}{\operatorname{argmax}} [s_{split}(i, j, k)] \quad (2)$$

のように決定される。スコア関数 $s_{split}(i, j, k)$ はスパンのベクトル表現 s_{ij} 、重みパラメータ v_s, W_s, b_s を用いて

$$s_{span}(i, j) = v_s \cdot \operatorname{ReLU}(W_s s_{ij} + b_s), \quad (3)$$

$$s_{split}(i, k, j) = s_{span}(i, k) + s_{span}(k, j) \quad (4)$$

のように多層パーセプトロンに基づき、スパンに対する分割 (i, j, k) のスコアを計算する。

3.2.3 核性ラベルの推定

各スパン (i, j) の核性ラベル \hat{l} はスパンの分割点の情報を用いて、スコア関数 $s_{label}(i, j, k, l)$ に基づき、

$$\hat{l} = LABEL(i, j, k) = \underset{l}{\operatorname{argmax}} [s_{label}(i, j, k, l)] \quad (5)$$

のように計算される。スコア関数 $s_{label}(i, j, k, l)$ は、分割点と同様に、重みパラメータ v_l, W_l, b_l とスパン (i, j) の分割点 k をもとに、多層パーセプトロンに基づき、二つのスパンのベクトル表現 s_{ik}, s_{kj} の間の核性のスコアを推定する。

$$s_{labels}(i, j, k) = v_l \cdot \operatorname{ReLU}(W_l [s_{ik}; s_{kj}] + b_l), \quad (6)$$

$$s_{label}(i, j, k, l) = [s_{labels}(i, j, k)]_l \quad (7)$$

3.2.4 分散表現

本節では、EDU、文、段落といった単語系列に対する固定長の分散表現と、それらの計算に必要な、系列中の単語に対する分散表現について説明する。

系列中に含まれる各単語 w_t の分散表現は、ELMo[8]と GloVe[7]により計算される。GloVe と異なり、ELMo は複数のベクトルを組み合わせることにより単語ベクトルが構成されるため、提案手法におけるベクトルの組み合わせ方法について説明する。単語 w_t に対する ELMo ベクトル r_t は、単語内の文字を畳み込む層と、二層の双方向 LSTM の計 3 層の内部状態 $\{h_{t,l}^{LM}\}_{l=1}^3$ により、ELMo ベクトル全体をスケールン

グする係数 γ^{LM} と、各層の正規化された重みである学習可能なパラメータ $\{s_l^{LM}\}_{l=1}^3$ を用いて以下のように計算される。

$$r_t = \gamma^{LM} \sum_{l=1}^3 s_l^{LM} h_{t,l}^{LM} \quad (8)$$

単語系列に対する分散表現は SIF[1] を用いて計算される。SIF は、単語の出現頻度に基づいた単語分散表現の加重平均により、文や段落といった単語の系列に対する分散表現を求める手法である。本研究において、 N 単語からなる系列 $C = (w_1, \dots, w_N)$ の分散表現 v_c は、GloVe ベクトル e_t と ELMo ベクトル r_t を結合した各単語 w_t の分散表現 $[e_t; r_t]$ と、単語の出現確率 $p(w_t)$ 、パラメータ a を用いて

$$v_c = \sum_{w_t \in C} \frac{a}{p(w_t) + a} [e_t; r_t] \quad (9)$$

のように計算される。パラメータ a はハイパーパラメータの定数 α と総単語数 Z を用いて

$$a = \frac{1 - \alpha}{\alpha Z} \quad (10)$$

のように計算され、頻出する単語に対して小さな値を取り、頻出する単語の重みを減少させる役割を担う。

3.2.5 目的関数

提案するトップダウン解析手法では、動的計画法を用いるようなボトムアップ解析手法とは異なり、全スパン分割の組み合わせに対する確率値の計算は困難である。そのために、提案手法では貪欲法による談話構造木の構築を前提とし、学習にはマージン最大化学習を選択した。また、2つのスパンの核性ラベル推定についてもスパン分割と同様に学習を行った。スパン分割及び核性ラベル推定の目的関数は、

$$\max(0, 1 + s_{split}(i, k^*, j) - s_{split}(i, \hat{k}, j)), \quad (11)$$

$$\max(0, 1 + s_{label}(i, j, k, l^*) - s_{label}(i, j, k, \hat{l})) \quad (12)$$

のように定義される。

3.3 関係ラベルの推定

提案手法において、スパン間の関係ラベルの推定は、二段階の談話構造解析手法[10]と同様、スパン分割と核性ラベル推定を行った後に実行される。スパン (i, k) と (k, j) の関係ラベル \hat{q}_{ijk} は、二つのスパンの表現ベクトル s_{ik} , s_{kj} 、二つのスパンの核性²の埋め込みベクトル n_{ijk} 、クロネッカーのデルタ δ^3 を用いて、

$$\hat{q}_{ij} = \underset{q}{\operatorname{argmax}} [\delta_q \cdot \operatorname{softmax}(W[s_{ik}; s_{kj}; n_{ijk}])] \quad (13)$$

²核性は N-N, N-S, S-N の 3 通りが存在する。提案手法では 10 次元ベクトルへの埋め込みを行った。

³添字が表す要素の次元のみ 1 となり、それ以外は 0 となる演算である。

により決定される。なお、式 (13) は、最尤推定に基づく点推定により学習される。

4 実験

4.1 実験設定

RST-DTB に含まれる Wall Street Journal の 385 記事を用いて提案手法の評価を行った。Heilman ら [3] の分割に従い、307 記事を学習データ、40 記事を開発データ、38 記事をテストデータとした。評価には従来手法と同様に、スパン (S)、核性 (N)、関係ラベル (R) の F_1 値を用いた。さらに二分木化による評価の偏りを低減する [6] ために、Parseval- F_1 値についても評価した。

提案手法の学習は勾配クリッピング 5.0 と重み $1e^{-4}$ の L2 正則化の下で Adam を使用して行い、学習率は 0.001 に設定した。50 epoch まで学習し、開発データで最も学習誤差の少ない epoch のモデルをテストに用いた。LSTM の隠れ層の次元数は 250 とし、Dropout 率は 0.4 とした。

提案手法 (Doc-Par-Sent-EDU) に対する比較手法としては、考慮する階層を EDU のみにした手法 (Doc-EDU)、EDU と段落にした手法 (Doc-Par-EDU)、EDU と文にした手法 (Doc-Sent-EDU) を用いた。さらに、文脈を考慮した単語ベクトルの影響を分析するために、提案手法から ELMo により獲得されるベクトルを除外した手法との比較も行った。また、提案手法が実用的な精度を達成できているかを判断するために、現在最高精度である Wang らの手法の他に、近年の手法をまとめている論文 [6] を参考に、各評価尺度において最高精度近辺の精度を達成可能な手法についても比較手法とした。正当な比較のために、Wang らの実験結果は Heilman らのデータ分割に基づいて再学習して得られた値で比較する。

4.2 実験結果

実験結果を表 1, 表 2, 表 3 に示す。この結果から提案手法はスパンの解析精度に関しては、現在の最高精度と同等の精度を達成していることが分かる。Wang らの手法は人手で考案した様々な特徴を組み合わせで利用しているが、提案手法は単語埋め込みベクトルに基づく単純な素性ベクトルしか利用していない。それにもかかわらず、これだけの性能を達成したことは提案手法の解析法としての能力の高さを示している。これは、ELMo を利用しない場合の精度の低下もわずかなことからも分かる。

一方で、核性ラベル推定の精度は最高精度の解析結果に及んでいない。これは、核性が下位のスパンの影響を受けやすく、ボトムアップの解析手法ではこの点を考慮しやすいこと、関連して、明示的な構文情報を利用していないことが原因であると考えられる。構文情報を利用すること自体は簡単にできるが、談話構造解析の前処理に構文解析を行うことが必須となると言

Model	S	N	R
Doc-EDU	84.8	69.7	55.9
Doc-Par-EDU	87.6	73.1	59.3
Doc-Sent-EDU	83.9	70.0	55.8
Doc-Par-Sent-EDU	87.8	73.7	60.0
(w/o ELMo)	86.8	68.7	54.7
Wang et al. (2017)	88.1	75.2	63.2
FH14gCRF	87.0	74.1	61.1
JE14 DPLP	85.0	71.6	62.0
Human	89.6	78.3	66.7

表 1: 各手法のマクロ平均 F_1 値

Model	S	N	R
Doc-EDU	81.4	65.1	50.02
Doc-Par-EDU	85.2	69.5	54.7
Doc-Sent-EDU	80.7	64.8	50.3
Doc-Par-Sent-EDU	85.4	70.0	55.0
(w/o ELMo)	84.5	64.5	50.1
Wang et al. (2017)	85.6	71.6	59.3
FH14gCRF	84.3	69.4	56.9
JE14 DPLP	82.0	68.2	57.8
Human	88.3	77.3	65.4

表 2: 各手法のマикро平均 F_1 値

語依存性が高まる可能性もある。今後、構文情報を利用せずに性能を向上させることが課題である。

関係ラベル推定の精度も核性ラベルの場合と同様に、Wang らの手法に及んでいない。この原因として、関係ラベル推定はスパン分割と核性ラベル推定の後に行うことから核性ラベル推定の精度が影響する、関係ラベル推定に単純な二層の多層パーセプトロンを利用していることが考えられる。ELMo を利用しないと精度が大きく低下していることから文脈情報の重要性が分かるので、これをさらに活用できるような分類モデルを考案することが課題である。

表 3 の結果は、表 1, 2 の結果よりもスコアが大きく劣っているが、これは手動で付与した EDU を入力としているため、Marcu らの評価手法において葉のスパンが必ず正解することによるものと考えられる。スコアの違いはあるものの手法間の優劣は表 1, 2 とは変わらない。

また、解析対象となる階層を変更した場合の結果から、提案手法である、談話構造木を階層に分割して解析する手法が、トップダウンの談話構造解析の精度向上に大きく寄与することが分かる。さらに、Doc-EDU, Doc-Par-EDU, Doc-Sent-EDU の結果の比較から、段落情報がスパン、核性、関係のいずれにおいても重要な情報であることが読み取れる。

5 まとめ

本論文では、EDU, 文, 段落といった、文書中に普遍に存在している階層構造に着目した、トップダウンの談話構造解析手法を提案した。

提案手法は RST-DTB を用いた実験で、スパンの評価において現在最高精度の談話構造解析器と同等の精度を達成した。また階層構造を考慮することにより、スパン、核、関係のいずれの評価においても解析精度

Model	S	N	R
Doc-EDU	63.5	50.9	41.4
Doc-Par-EDU	70.8	57.9	45.7
Doc-Sent-EDU	61.9	51.6	41.7
Doc-Par-Sent-EDU	71.4	58.2	46.0
(w/o ELMo)	69.4	51.7	41.8
Wang et al. (2017)	71.7	60.4	50.2
FH14 gCRF	68.6	55.9	45.8
JE14 DPLP	64.1	54.2	46.8
human	78.7	66.8	57.1

表 3: Parseval- F_1 値

が改善されることを示した。さらに、階層構造の中では、特に段落情報がトップダウンの談話構造解析において重要な情報であることを示した。

一方で、現在の提案手法では、核性ラベル推定をスパンの解析と同時にやっているため、葉ノードからの核性ラベルの遷移がその後の核性ラベルの推定に影響しているにもかかわらず、その情報を推定に利用できていない。関係ラベルと同様に、核性ラベルについても木構造の決定後に推定を行うことで、解析精度の向上を目指すことを今後の課題としたい。

参考文献

- [1] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. A simple but tough-to-beat baseline for sentence embeddings. In *ICLR*, 2017.
- [2] Lynn Carlson, Daniel Marcu, and Mary Ellen Okurovsky. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *SIGDIAL Workshop*, 2001.
- [3] Michael Heilman and Kenji Sagae. Fast rhetorical structure theory discourse parsing. *CoRR*, Vol. abs/1505.02425, , 2015.
- [4] Qi Li, Tianshi Li, and Baobao Chang. Discourse parsing with attention-based hierarchical neural networks. In *EMNLP*, 2016.
- [5] W.C. Mann and S.A Thompson. Rhetorical structure theory: A theory of text organization. Technical Report ISI/RS-87-190, USC/ISI, 1987.
- [6] Mathieu Morey, Philippe Muller, and Nicholas Asher. How much progress have we made on rst discourse parsing? a replication study of recent results on the rst-dt. In *EMNLP*, 2017.
- [7] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.
- [8] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke S. Zettlemoyer. Deep contextualized word representations. In *NAACL-HLT*, 2018.
- [9] Mitchell Stern, Jacob Andreas, and Dan Klein. A minimal span-based neural constituency parser. In *ACL*, 2017.
- [10] Yizhong Wang, Sujian Li, and Houfeng Wang. A two-stage parsing method for text-level discourse analysis. In *ACL*, 2017.
- [11] Nan Yu, Meishan Zhang, and Guohong Fu. Transition-based neural rst parsing with implicit syntax features. In *COLING*, 2018.