談話構造を考慮する階層的注意機構による抽出型ニューラル単一文書要約

石垣 達也[†] 上垣外 英剛[‡] 高村 大也[‡§] 奥村 学[‡]
[†]東京工業大学総合理工学研究科 [‡]東京工業大学科学技術創成研究院 [§]産業総合技術研究所
{ishigaki, kamigaito}@lr.pi.titech.ac.jp {takamura, oku}@pi.titech.ac.jp

1 はじめに

本稿では，原文書の談話構造を考慮する新たな抽出型ニューラル要約モデルを提案する．なお，本研究において抽出型要約課題は，原文書の各文に対し抽出すべきか否かを表現する二値のラベルを付与する，系列ラベリング問題として定式化する．

修辞構造理論 [7] に代表される談話構造を表現する枠組みは，文書中の文や単語などの間に内在する意味的なつながりに着目する．本研究では特に文間の意味的なつながりを考える．図 1 に原文書とその談話構造の例を示す．$S1$ と $S2$ は原因とその結果について言及している．$S3$ は $S2$ に補足的な情報を与え，さらに $S4$ は $S3$ を補足する．このような談話構造は抽出型要約において文の重要度を算出するための手がかりとなる．例えば，原因よりも結果，補足する文よりも補足される文が重要とすれば，$S2$ を要約に含めることで重要な情報を含んだ要約を出力できる．

整数計画法に基づく要約手法において，談話構造解析器の出力する談話構造を考慮しながら文選択することで，情報量および一貫性の観点から要約器の性能が向上することが報告されている [5]. これらの要約器は前処理として談話構造解析器を利用するため，その解析誤りの影響を大きく受ける．談話構造解析器は長いテキストや，学習に用いられたドメインとは異なるドメインのテキストに適用すると性能が大きく劣化する．

一方で，リカレントニューラルネットワーク (RNN) に基づく抽出型手法が 2016 年以降，単一文書要約において良い性能を示している．この手法は原文書を文の系列とみなしベクトル化し文の重要度を決定し，文間の談話構造は明示的には利用しない．談話構造に関する情報の欠如は，重要度スコア決定における性能劣化や出力要約の一貫性の低下を引き起こす可能性がある．

そこで本研究では，談話構造解析器の解析誤りによる影響を抑えながら，RNN を用いた要約モデルの性能における利点を活用するため，原文書の談話構造と文の重要度スコアリング器を同時に学習する新たな枠組みを提案する．本モデルにおいて，原文書の談話構造は階層的な注意機構として表現される．スコアリング器は RNN に基づくデコーダとソフトマックス関数により構築され，対象文に加えその親の情報も考慮しながら，対象文の重要度スコアを算出する．

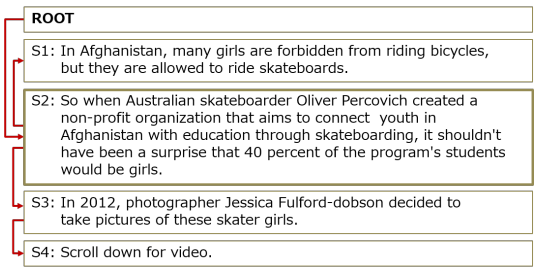

図 1: 談話構造の例

DailyMail データセットを用いた評価実験において，提案手法がベースラインよりも ROUGE 値および人手評価において良い評価値を得た．さらに，既存の性能の良い手法と同等もしくは，より良い結果を得た．

2 RNN に基づく抽出型要約モデル

本節では初めに，多くの既存手法で採用されている RNN に基づく抽出型要約モデルの基本構成について説明する．この基本構成における要約器は文エンコーダ，文書エンコーダおよび文スコアリング器から構成される [1, 8, 2]．以下に順にその動作について説明する．

文エンコーダ: 文エンコーダの目的は N 文から構成される原文書 $\mathbf{x}$ 中の文 x_i $(0 \leq i \leq N)$ を文埋め込み表現 h_i に変換することである．ここで，x_0 は談話構造木における根を表現する特別な文であり，本稿では ROOT と呼ぶ．文エンコーダは初めに，x_i 中の各単語を単語埋め込み表現に変換する．x_i の n 番目の単語の単語埋め込み表現を $emb(w_{i,n})$ とする．単語を両方向 Long Short-term Memory (LSTM) で読み込むと $\overrightarrow{e}_{i,n} = \mathrm{LSTM}(\overrightarrow{e}_{i,n-1}, emb(w_{i,n}))$ および $\overleftarrow{e}_{i,n} = \mathrm{LSTM}(\overleftarrow{e}_{i,n+1}, emb(w_{i,n}))$ となる．$\overleftarrow{e}_{i,n}$ および $\overrightarrow{e}_{i,n}$ はさらに一つのベクトルに結合され，$h_{i,n} = [\overrightarrow{e}_{i,n}; \overleftarrow{e}_{i,n}]$ を得る．すべての単語に対する $h_{i,n}$ を平均したベクトルを最終的な文埋め込み表現 h_i とする．

文書エンコーダ: 文書エンコーダは文脈を考慮した文表現 H_i および文書全体を表現するベクトル K を生成する．具体的には，両方向 LSTM を用い，$\overrightarrow{H}_i =$

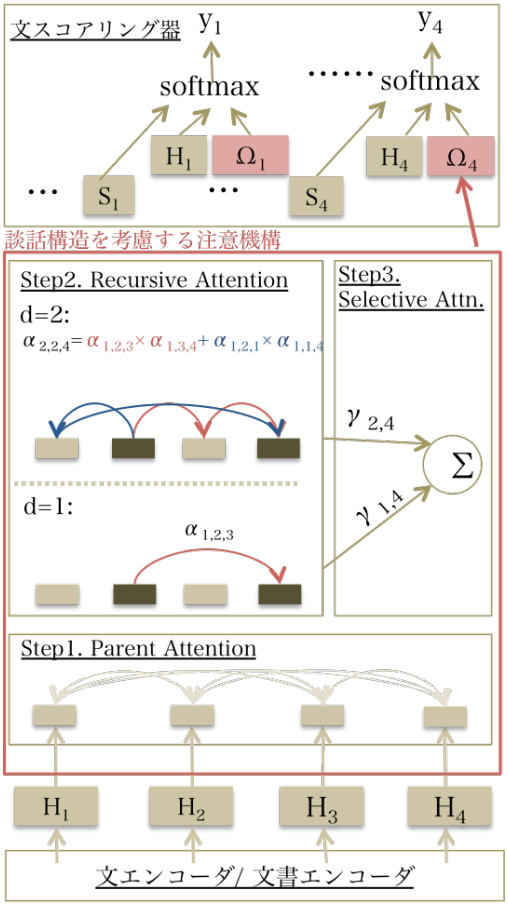

図 2: 注意機構の動作概要．Ω_4 の生成過程を示す．

$\mathrm{LSTM}(\overrightarrow{H}_{i-1}, h_i)$ および $\overleftarrow{H}_i = \mathrm{LSTM}(\overleftarrow{H}_{i+1}, h_i)$ を生成し，結合する: $H_i = [\overrightarrow{H}_i; \overleftarrow{H}_i]$．すべての文に対し文脈を考慮した文表現 ($\mathbf{H} = \{H_1, ..., H_N\}$) を獲得し，$H_i$ の平均ベクトルを文書表現 K とする．

文スコアリング器: 文スコアリング器は文 $x_t (1 \leq t \leq N)$ を要約に含める確率 $p(y_t = 1|\boldsymbol{x}, \theta)$ を，LSTM によるデコーダとソフトマックス関数により出力する．デコーダは，時点 t $(1 \leq t \leq N)$ において，直前の時点での LSTM の出力 s_{t-1} と h_t を受け取り，新たなベクトルを出力する: $s_t = \mathrm{LSTM}(s_{t-1}, h_t)$. なお，$s_0$ は文書エンコーダの後ろ向き LSTM の最終状態 $\overleftarrow{H}_0$ とする．文 x_t を要約に含める確率は，文脈を考慮した文表現ベクトル H_t，LSTM の出力 s_t および文書表現 K を結合し以下の式により得る:

$$p(y_t = 1|\mathbf{x}, \theta) = \delta_{y_t=1} \cdot \mathrm{softmax}(\mathbf{W}_o[H_t; s_t; K]). \quad (1)$$

ここで，W_o は重み行列，$\delta_{y_t=1}$ は $y_t = 1$ に対応する次元のみ 1 を取るベクトルで，ソフトマックス関数の出力との内積を取り，文 x_t を要約に含める確率を得る．

3 談話構造を考慮する注意機構

本研究では式 (1) のソフトマックス関数に x_i の親を表現するベクトル Ω_i を追加で入力することで，談話構造を考慮しながら確率を計算するモデルに拡張する．親を表現するベクトルの獲得には，単語間 の依存構造を捉える Kamigaito ら [6] のモデルを文間の談話依存構造を捉えるよう拡張して用いる．図 2 に x_4 の親を表現するベクトル Ω_4 が生成される過程を示す．3.1 節で生成過程を説明し，3.2 節で定式化する．

3.1 親を表現するベクトルの生成過程

Step1: Parent Attention Parent Attention はすべての文の組み合わせ x_k および x_j（ただし $k \neq j$）について x_k が x_j の親となる確率 $p(k|j, \mathbf{H})$ を計算する．図 2 の Step1 に示すように，$\mathbf{H}$ に含まれる各ベクトルが頂点に対応するグラフを考える．矢印の始点を親，終点を子とすると，$\mathbf{H}$ は $p(k|j, \mathbf{H})$ を H_j に対応する頂点から H_k に対応する頂点に向かう辺の重みとした重み付き全結合グラフで表現される．

Step2: Recursive Attention Recursive Attention は Parent Attention が生成した全結合グラフの情報を用い，x_4 の d 次親を表現するベクトル $\gamma_{d,4}$ を，$\mathbf{H}$ を加重平均することで得る．ここで，2 頂点を結ぶ経路に含まれる辺の数を 2 頂点間の距離 d と考える．2 頂点間の距離が d である親子関係の親を d 次親と呼ぶ．

Recursive Attention は 1 次親 ($d = 1$) の場合から計算を始める．各 2 文の組について距離が 1 である親子関係が成り立つ確率は Parent Attention により計算済みである．Recursive Attention はこの確率を H_k(ただし $k \neq 4$) に与える重み $\alpha_{1,k,4}$ とし，加重平均ベクトル $\gamma_{1,4}$ を得る．

$d = 2$ の場合の H_k に与える重み $\alpha_{2,k,4}$ の計算過程について述べる．この場合は x_4 と距離が 2 離れた親子関係を考える．図 2 において，x_2 から x_4 に向かう経路は赤色で示す線および青色で示す線の 2 通りがある．Recursive Attention は H_k に対応する頂点から H_4 に対応する頂点に向かう経路に含まれる辺の重みを掛け合わせ，すべての経路について足し合わせた確率値を H_k に対しての重みとして扱う．すなわち，$\alpha_{2,2,4} = \alpha_{1,2,3} \times \alpha_{1,3,4} + \alpha_{1,2,1} \times \alpha_{1,1,4}$ である．すべての H_k(ただし $k \neq 4$) に対して，重み $\alpha_{2,k,4}$ を計算したら，これらの重みで H_k を加重平均し 2 次親を表現するベクトル $\gamma_{2,4}$ を得る．

$d > 2$ に対しても，$d = 2$ までに計算した重みを用いて，3.2 節に示すように再帰的に重みを計算する．

Step3: Selective Attention Selective Attention は d 次親を表現するベクトル $\gamma_{d,4}$ が得られたら，さらにそれらのベクトルに対する重みを計算する．計算した重みを用いて加重平均したベクトル Ω_4 を式 (1) に追加で入力し，文 4 を要約に含めるスコアを算出する．

3.2 定式化

談話構造を考慮する注意機構を定式化する．Parent Attention はすべての文の組み合わせ x_k および x_j に対し (ただし $k \neq j$) 親子となる確率を計算する：

$$p(k|j, \mathbf{H}) = \sigma(g(k, j)), \quad (2)$$

$$g(k, j) = v_a \cdot \tanh(U_a H_k + W_a H_j). \quad (3)$$

ここで，v_a は重みベクトル, U_a および W_a は重み行列である．

Recursive Attention は再帰的に x_k が x_j の d 次親となる確率 $\alpha_{d,k,j}$ を計算する:

$$\alpha_{d,k,j} = \begin{cases} p(k|j, \mathbf{H}) & (d = 1), \quad (4) \\ \sum_{l=1}^{N} \alpha_{d-1,k,l} \times \alpha_{1,l,j} & (d > 1). \quad (5) \end{cases}$$

さらに，以下の制約を $\alpha_{1,k,i}$ に課す:

$$\alpha_{1,k,j} = \begin{cases} 1 & (k = 0, j = 0), \quad (6) \\ 0 & (k > 0, j = 0), \quad (7) \\ 0 & (k = i, j \neq 0). \quad (8) \end{cases}$$

式 (7) および式 (8) は ROOT から出るノードを ROOT 自身に限定し，ROOT が親に持たないよう制約する．一方で，ROOT 以外の文は親を持つので式 (8) により自分自身への辺は持たないよう制約する．計算された確率 $\alpha_{d,k,j}$ を重みとして用いて $\mathbf{H}$ の加重平均ベクトル $\gamma_{d,j}$ を以下のように得る:

$$\gamma_{d,j} = \sum_{k=0}^{N} \alpha_{d,k,j} H_k. \quad (9)$$

全ての文の組み合わせに関し、x_j の d 次親を表現するベクトル $\gamma_{d,j}$ が得られたら，Selective Attention はデコーダの現在の時点 t における、それぞれの d に対応する $\gamma_{d,t}$ を重み付け，加重平均ベクトル Ω_t を得る:

$$\beta_{d,t} = \delta_d \cdot \mathrm{softmax}(\mathbf{W}_\beta [H_t; s_t; K]), \quad (10)$$

$$\Omega_t = \sum_d \beta_{d,t} \cdot \gamma_{d,t}. \quad (11)$$

ここで δ_d は特定の d に対応する次元のみ 1 を取るベクトルで，ソフトマックス関数の出力した確率分布を確率に変換する．

談話構造を考慮するよう式 (1) を拡張した以下の関数により，x_t を要約に含める確率を出力する:

$$p(y_t = 1|\mathbf{x}, \theta) = \delta_{y_t=1} \cdot \mathrm{softmax}(\mathbf{W}_o [H_t; s_t; K; \Omega_t]). \quad (12)$$

3.3 目的関数

重み行列の重みは以下の目的関数を最小化するよう更新する：

$$-\log p(\mathbf{y}|\mathbf{x}) - \lambda \cdot \sum_{k=1}^{N} \sum_{i=1}^{N} E_{k,i} \cdot \log \alpha_{1,k,i}. \quad (13)$$

この式は第 1 項で訓練データ中の正解ラベルアノテーションを再現するようにパラメータを更新する．第 2 項において，$E_{k,i}$ は訓練事例において x_k から x_i への辺が存在する場合に 1 をとる二値関数である．よって，正解の談話構造アノテーションを再現するよう $\alpha_{d,k,i}$ を学習する，談話構造解析のための項になっている．

4 実験

文書要約の既存研究では固有名詞を @entity に置き換えた Cheng ら [1] による前処理済み DailyMail データセットがしばしば用いられる．しかし，このデータでは談話構造解析器の必要とする段落境界や単語の表層情報が欠如しており，解析性能が著しく劣化する．そのため，本研究では前処理済みデータではなく Hermann ら [3] の DailyMail データセットに対し，HILDA パーザ [4] を用いて談話構造情報を自動アノテーションしたデータを用いる．DailyMail データセットには新聞記事と人間が記述した “ハイライト” が含まれており，ハイライトを生成的に作られた正解要約とみなす方法がいくつかの既存研究で採用されている．抽出型要約器の訓練には抽出文か否かを示す二値のアノテーションが必要である．そのため，本研究では Nallapati ら [8] のデータ作成手法に基づき，ハイライトを参照要約とし ROUGE-2 F1 値を最大にする文集合を抽出すべき文集合として自動的にアノテーションした．

比較手法として，既存の抽出型ニューラル要約器として良い性能を示している SummaRuNNer [8]，談話構造を考慮しない式 (1) を確率の計算に用いるベースライン（no-attn）および先頭 3 文を選ぶ LEAD-3 を用いる．これらの手法と ROUGE-1，ROUGE-2，ROUGE-L を 75 バイト，275 バイトおよび参照要約長の 3 つの出力長制約で比較する．また，無作為に選択した 100 文書に対し 5 人の評価者が一貫性および情報量について良い順に並べ替える人手評価も行った．この評価では評価者が同順位を付与することも許容した．評価者の募集は Amazon Mechanical Turk で行った．

5 結果

表 1 に各手法の ROUGE 値を示す．談話構造を考慮する注意機構を持たないモデル no-attn と提案する注意機構を追加したモデル DIS を比較すると，すべての出力長制約において，各 ROUGE 値が向上した．よって，談話構造を考慮することで ROUGE 値が向上することがわかった．既存のニューラル要約手法として良い性能を示している SummaRuNNer との比較においては，出力長制約 275 バイトでの ROUGE-1 以外

	75			275			Ref.		
	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L
DIS, d={1}	22.9	9.6	12.1	41.9	17.4	**35.2**	41.1$^+$	**16.9**$^+$	36.7$^+$
DIS, d={1,2}	**25.0**$^+$	11.1$^+$	13.4$^+$	41.7	17.4	35.0	41.0	16.7	36.7$^+$
DIS, d={1,2,3}$^\#$	24.6$^+$	**11.2**$^+$	**13.5**$^+$	41.8	17.2	**35.2**	41.1$^+$	16.8$^+$	36.8$^+$
DIS, d={1,2,3,4}	24.1$^+$	10.8$^+$	13.2$^+$	41.1	**17.5**	35.1	**41.2**$^+$	**16.9**$^+$	**37.0**$^+$
Lead-3	23.0	9.4	11.8	41.9	17.0	32.5	40.4	16.3	36.1
SummaRuNNer (re-trained)	23.2	9.6	11.0	**42.0**	17.2	32.5	37.6	14.8	33.7
no-attn	20.1	7.1	10.4	39.6	15.4	33.3	39.3	15.3	35.2

表 1: DailyMail データセット (non-anonymized) における ROUGE 値．[+] ROUGE スクリプトの-c オプションにより統計的有意であることを確認.

	75(一貫性/情報量)				275(一貫性/情報量)				正解要約長 (一貫性/情報量)			
	1	2	3	4	1	2	3	4	1	2	3	4
DIS	.24/.22	**.31**/.24	.23/.15	.23/**.38**	.33/.26	**.35**/.24	.23/**.27**	.10/23	**.40/.37**	.35/.30	.21/.24	.03/.09
LEAD-3	**.42**/22	.25/.24	.23/**.30**	.10/.24	.25/.23	**.35**/.26	.30/**.28**	.10/.23	.30/.23	.31/**.31**	**.34/.31**	.05/.14
SummaRuNNer	.19/**.38**	.27/.33	**.38**/.17	.14/.11	**.35/.28**	.23/**.28**	.25/.25	.18/.19	.27/.21	.27/.30	**.32/.32**	.16/.16
no-attn	.14/.16	.16/.18	.16/**.38**	**.53**/.27	.08/.23	.08/.23	.23/.20	**.63/.34**	.05/.18	.06/.09	.12/.12	**.77/.60**

表 2: 人間によるランキング．数値は各手法が 1–4 位に評価された割合を示す．スラッシュ(/) の左が一貫性，右が情報量に関する評価である．

では，提案手法が良い性能を示した．さらに 75 バイトおよび参照要約長の出力長制約においては，提案手法がほとんどのパラメータにおいて他のベースラインを統計的に有意に上回る ROUGE 値を獲得した．75 バイト，参照要約長，275 バイトの出力長の設定の順に，提案手法と SummaRuNNer の性能差が小さくなった．これは，出力長が短い設定においては提案手法がより良い性能を示し，長い出力長制約の設定については提案手法も SummaRuNNer も同等の性能に収束しているためと考えられる．よって，談話構造の導入は特に短い出力長制約下で有効であることが示唆される．

表 2 に人手評価の結果を示す．表中の数値は，並べ替え評価において，各手法が 1–4 位に判定された割合である．75 バイトの要約長制約の設定において，Lead-3 が最も良い性能を示した．Lead-3 は連続する 3 文を出力する性質を持つが他の比較手法はこの性質を持たないためだと考えられる．提案手法 DIS は Lead-3 の次に良い評価を得た．SummaRuNNer と no-attn よりも 75 バイト設定において一貫性について良い評価を得ていることから，談話構造が寄与していることがわかる．長い要約長制約である 275 バイトの設定下では，DIS，Lead-3，SummaRuNNer がそれぞれ最も良いと判定された割合は，0.33，0.25，0.35 で，2 番目に良いと判定された割合は 0.35，0.35，0.23 と，近い性能を示した．正解要約長の制約下においては DIS が他の手法よりも良い評価を得た．これらは，ROUGE 値の傾向と同様である．

6 おわりに

本研究では原文書の談話構造を捉える新たな注意機構を提案した．評価実験において提案注意機構の導入による ROUGE 値の向上を確認した．人手評価においても既存ニューラル手法と同等か良い性能を示した．

参考文献

[1] Jianpeng Cheng and Mirella Lapata. Neural summarization by extracting sentences and words. In *Proceedings of ACL2016*, pp. 484–494, Berlin, Germany, 2016.

[2] Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of NAACL2018*, pp. 615–621, 2018.

[3] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Proceedings of NIPS2015*, pp. 1693–1701, 2015.

[4] Hugo Hernault, Helmut Prendinger, Mitsuru Ishizuka, et al. Hilda: A discourse parser using support vector machine classification. *Dialogue & Discourse*, Vol. 1, No. 3, 2010.

[5] Tsutomu Hirao, Yasuhisa Yoshida, Masaaki Nishino, Norihito Yasuda, and Masaaki Nagata. Single-document summarization as a tree knapsack problem. In *Proceedings of EMNLP2013*, pp. 1515–1520, 2013.

[6] Hidetaka Kamigaito, Katsuhiko Hayashi, Tsutomu Hirao, and Masaaki Nagata. Higher-order syntactic attention network for longer sentence compression. In *Proceedings of NAACL2018*, pp. 1716–1726, 2018.

[7] William C. Mann and Sandra A. Thompson. Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, Vol. 8, No. 3, pp. 243–281, 1988.

[8] Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of AAAI2017*, pp. 3075–3081, 2017.