生成要約のための隣接畳み込みニューラルネットワーク

近藤 雅芳　　松本 裕治
奈良先端科学技術大学院大学 情報科学研究科
理化学研究所 AIP - 知識獲得グループ

{kondo.masayoshi.ki1, matsu}@is.naist.jp

1 はじめに

自然言語処理分野へのニューラルネットワークの適用は目覚しい成果を挙げている．とりわけ、大きく進展している研究のひとつに自動要約の研究がある．自動要約は、与えられた文章に対して予め定められた最大文字数内でその要旨をまとめた文章を構成し、出力するシステムの研究である．

自動要約は、抽出要約と生成要約の二つの研究アプローチに大別できる．抽出要約は、与えられた文章内の表現を取り出し、それらの並び替えを行うことで要約文を構成する．抽出要約は与えられた文章の表現を用いることで可読性が高い要約文が得られる反面、その表現は与えられた文章の内容の範囲に留まる．これに対して、生成要約はシステムを用いて要約文を直接生成する手法であり、与えられた文章には存在しない言い換えといった新しい表現を生成可能である点が抽出要約と異なる．

近年では、ニューラルネットワークを用いて逐次的に要約文を生成する手法が従来の抽出要約と同程度以上の精度を実現できる報告がなされ、活発に研究が行われている [4, 9, 10, 11, 14]．例えば、Rush ら [10] は、ニュース記事の冒頭文章と記事見出しを原文と要約文のペアとした大規模な短文要約データセットを作成し、CNN による短文要約生成の手法を提案した．また、複数文を入出力とする長文要約では、Nallapati らの研究 [9] と See らの研究 [11] がある．Nallapati らは、BiLSTM をエンコーダとする attention-seq2seq モデル [2] に copy 機構を組み込むことで attention 機構のみの場合と比べて低頻度語彙の出力間違いを減少させ、精度向上ができることを示した．See らは、要約生成時の単語や文章の繰り返し生成を抑制するために coverage 機構を組み込むことで、繰り返し生成による間違いを低減させ精度を向上させた．

しかしながら、ニューラルネットワークを用いた生成要約にはいくつかの課題がある．その課題の一つに、長文のエンコードの困難さが挙げられる．従来のニューラル生成要約では、seq2seq モデル [2, 13] を用いることが一般的である．このため、長文要約のような長期系列を入力とする場合、学習時の勾配消失と爆発の問題は避けられない．近年では、このような問題と学習の高速化を目的に新しい非再帰型ニューラルネットがいくつか提案されている [5, 6, 12, 15] が、これらはパラメータを非常に多く必要することや、系列情報を多層化することで暗に考慮するために長期系列に対しては多くの層数が必要となり、結果として大きな計算資源を要する、といった欠点が残っている．

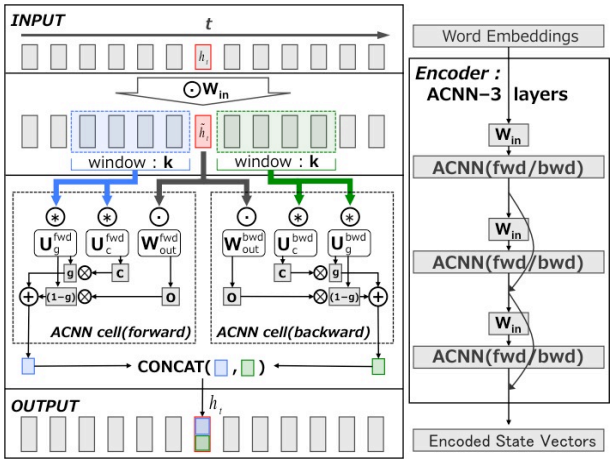

図 1: 提案手法の ACNN と本研究で用いるエンコーダ構成．

本研究では、このような問題を解消するために、**隣接畳み込みニューラルネットワーク (Adjoint Convolutional Neural Network; ACNN)** を提案する．ACNN は、明示的に系列情報を考慮しながら入力系列に対して並列処理を可能にする新しい非再帰型ニューラルネットワークである．また、ACNN 内の重みの一部を層間で共有することでモデルパラメータ数を削減し、先行研究の最高精度を示す BiLSTM エンコーダの 42 %程度のパラメータ数で、同程度以上の精度を得られることを長文要約データセットを用いた実験により示す．本研究の貢献事項を以下にまとめる．

- 隣接畳み込みニューラルネットワーク (ACNN) を提案する．ACNN は、系列情報を考慮しながら並列処理を可能とする新しい非再帰型ニューラルネットワークである．
- CNN/DailyMail データセットを用いた長文生成要約タスクでその有効性を検証し、ACNN が先行研究の最高精度である BiLSTM エンコーダと同程度以上の精度であることを示す．
- 重みの層間共有により、ACNN は先行研究の BiLSTM の約 42%程度のパラメータ数にまで抑えられ、従来の非再帰型ニューラルネットに比べ、小さいパラメータ数で高いパフォーマンスを示す．

本稿の構成を述べる．第 2 章でニューラル生成要約について述べた後、第 3 章で提案モデルについて説明を行う．第 4 章では実験とその評価を行い、第 5 章で結びを述べる．

2 ニューラル生成要約

ニューラル生成要約は、長さ l_x の文章 X が与えられたとき、文章 X より短い長さ $l_y(< l_x)$ で構成される要約文 Y をニューラルネットを用いて獲得することを目的とする．X と Y は入力文章と要約文を意味し、共通の語彙集合 V からなる単語順序列で表される．本稿では以降、X と Y をそれぞれ入力系列と出力系列と呼び、$X = \{x_i\}_{i=1}^{l_x}, Y = \{y_j\}_{j=1}^{l_y}$ $(x_i, y_j \in \{0,1\}^{|V|})$ と表記する．

一般的に、ニューラル生成要約では seq2seq モデル [2, 13] が用いられる．このモデルは、パラメータ Θ を備える以下のような 3 つの関数から構成される．

$$\begin{cases} H = Encoder(X; \Theta_{enc}) \\ \hat{y}_t = Decoder(H, \hat{Y}_{<t-1}, Z; \Theta_{dec}) \\ Z = OtherModule(H, \hat{y}_{t-1}; \Theta_{other}) \end{cases}$$

H と Z は、それぞれ入力系列 X に対するエンコード特徴量系列と、H と Y を入力とした付加的な機能を備えた関数から得られる特徴量を表す．

本研究では、ベースラインに See らのモデル [11] を用いる．彼らは、$Encoder$ と $Decoder$ にそれぞれ一層の BiLSTM と一層の LSTM を用いており、$OtherModule$ に attention 機構、pointer-generator 機構、coverage 機構の 3 つを用いて長文要約の精度検証を行った．pointer-generator 機構は、入力系列の単語と $Decoder$ による予測出力単語の双方を考慮して最終出力を行う仕組みであり、coverage 機構は単語の繰り返し生成を抑制する仕組みであるが、詳しい説明は紙面の都合上、割愛する．本研究では、$Encoder$ を提案モデルに変更して先行研究 [11] と同様の実験設定[1] で検証を行う．

2.1 モデルの学習

モデル $P(y_t|y_{<t-1}, X; \Theta)$ の学習を行う．学習は、まず初めに、複数の入出力ペア (X, Y) から構成される訓練データ D とモデルを用いて、以下の式から逐次的に予測系列 $\hat{Y} = \{\hat{y}_t\}_{t=1}^{l_{\hat{y}}}$ を得る．

$$\hat{y}_t = \underset{\hat{y}_t \in V}{\operatorname{argmax}} P(\hat{y}_t|\hat{y}_{<t-1}, X; \Theta) \quad (1)$$

次に、モデルから得た予測系列 $\hat{Y}$ を用いて、下記の目的関数の最小化することで行う．

$$L(Y, \hat{Y}) = \frac{1}{|D|} \sum_D \sum_j y_j \log \hat{y}_j \quad (2)$$

目的関数 L は、予測出力系列 $\hat{Y}$ と正解出力系列 Y のクロスエントロピー誤差関数を示す．

2.2 評価方法

評価には、テストデータを用いる．学習済みニューラルネットワークに対してテストデータの入力系列 X を与え、出力された予測系列 $\hat{Y}$ とテストデータの X に対する正解系列 Y とを以下の精度指標に基づいて評価を行う．

[1]先行研究 [11] の実験に関するスクリプト及び設定は、`https://github.com/abisee/pointer-generator` で公開されている．

ROUGE [8][2]: 正解系列と予測系列の N-gram 一致度に基づいた数値指標であり、一致度が高いほど大きな値を示す．本稿では、ROUGE-1/-2/-L の 3 つの指標で評価を行う．ROUGE-1/-2 は、正解系列と予測系列のユニグラムとバイグラムの一致度に基づいた指標であり、ROUGE-L は一致する最大部分系列長に基づいた指標である．

METEOR [3][3]: 正解系列と予測系列のユニグラム一致度に基づいた評価指標であり、一致度が高いほど大きな値を示す．本稿では、単語間の一致度に基づいた評価である exact match mode と、語幹や同義語、言い換えも含めた評価を行う full mode (+ stem/syn/para) で評価を行う．

3 隣接畳み込みニューラルネットワーク

隣接畳み込みニューラルネットワーク (Adjoint Convolutional Neural Network;ACNN) の説明を行う．提案モデルを図 1 に示す．ACNN は、2 つの特徴を備える．

1. 入力系列に対して、前方向と後方向の双方向情報を考慮しつつ、並列処理を可能とするモデル構造
2. 提案モデルを多層化した際に増加するパラメータ数を抑制するための重みの層間共有

3.1 モデルの構造

ACNN は、予め決められたコンテクスト窓を用いた畳み込み演算と行列積演算の組み合わせによって構成される．ACNN によるエンコード操作を以下の数式で表現する．

$$h_t^l = ACNN(h_t^{l-1}, h_{[:]}^{l-1}, \Theta^l) \quad (3)$$

h_t^l は、入力系列の t 番目の要素に対する l 層目の ACNN 関数のエンコード特徴量である．また、$h_{[:]}$ は、系列の部分要素列のエンコード特徴量行列を示す．具体的には、系列 h に対する位置 t を起点として後方向に $+k$ までを含む特徴量行列を $h_{[t:+k]}$ と表す．数式 (3) の $ACNN$ 関数は、パラメータ集合 $\Theta^l = \{W_{out}^l, U_c^l, U_g^l, b_{out}^l, b_c^l, b_g^l\}$ を用いて、具体的に以下のように計算される．

$$o_t^l = tanh(W_{out}^l \cdot h_t^{l-1} + b_{out}^l) \quad (4)$$
$$c_t^l = tanh(U_c^l * h_{[:]}^{l-1} + b_c^l) \quad (5)$$
$$g_t^l = \sigma(U_g^l * h_{[:]}^{l-1} + b_g^l) \quad (6)$$
$$h_t^l = g_t^l \otimes o_t^l + (1 - g_t^l) \otimes c_t^l \quad (7)$$

なお、$h_t^0 = e(x_t)$[4]であり、$\cdot$ は行列積演算、$*$ はコンテクスト方向への畳み込み積演算、$\otimes$ は 2 つの行列間の

[2]1.5.5 ver を利用した．評価時のコマンドオプションを次にそれぞれ記載する．ROUGE-1 : -a -m -n 1 -x / ROUGE-2 : -a -m -n 2 -x / ROUGE-L : -a

[3]1.5 ver を利用した．評価スクリプトは、`http://www.cs.cmu.edu/~alavie/METEOR` から入手できる．

[4]$e(x_t)$ は、x_t の単語埋め込み表現ベクトルを示す．

Encoder Model	ROUGE		
	1	2	L
Recurrent Model			
BiLSTM [11]	31.33	11.81	28.83
BiLSTM (re-exam)	33.24	13.11	29.68
Selective BiLSTM[16]	32.91	12.94	29.41
LSTM	32.96	12.83	29.31
Non-Recurrent Model			
FFNN	26.37	6.47	23.54
ACNN (proposed)	**33.65**	**13.41**	**30.38**

表 1: attention 機構のみを用いた場合における各エンコーダの ROUGE 値 (F-score)

要素積演算、σ はシグモイド関数を示す．各パラメータの大きさは、$W^l_{in} \in \mathbf{R}^{H\times 2H}, W^l_{out} \in \mathbf{R}^{H\times H}, U^l_c \in \mathbf{R}^{H\times K}, U^l_g \in \mathbf{R}^{H\times K}, b^l_{out}, b^l_c, b^l_g \in \mathbf{R}^H$ であり、H は特徴量の次元、K はコンテクスト窓の大きさを示す．

本研究で用いる提案モデルは、この ACNN を前方向と後方向で組み合わせて用いる．すなわち、前方向 cell と後方向 cell でそれぞれ独立なパラメータ集合 $\Theta^{fwd}, \Theta^{bwd}$ を用いて、h^l_t の計算は次のようになされる．

$$\tilde{h}^{l-1}_t = tanh(W^l_{in} \cdot h^{l-1}_t + b^l_{in}) \quad (8)$$

$$h^{l,fwd}_t = ACNN(\tilde{h}^{l-1}_t, \tilde{h}^{l-1}_{[t:-k]}, \Theta^{l,fwd}) \quad (9)$$

$$h^{l,bwd}_t = ACNN(\tilde{h}^{l-1}_t, \tilde{h}^{l-1}_{[t:+k]}, \Theta^{l,bwd}) \quad (10)$$

式 (8) は、前層のエンコード特徴量の大きさを変更する全結合ユニットである．h^l_t は、式 (9) と式 (10) を用いて以下のように得る．

$$h^l_t = concat[h^{l,fwd}_t : h^{l,bwd}_t] \quad (11)$$

$concat[\cdot:\cdot]$ は 2 つのベクトルの concatinate 演算を示す．上記の一連の式に示すように、同じ l 層にある各エンコード特徴量に対して系列の位置を示す変数 t の再帰的な式となっていない．このことから、提案モデルである ACNN は系列方向に対して再帰的な構造をもたず、入力系列に対して並列処理が可能となっている．

3.2 重みの層間共有

ACNN は層内に畳み込み演算と行列積演算を備えている．また、本研究では、ACNN cell とは別に前層の出力の次元を変更する重みを各層に一つ導入している．この時、行列積の重み W^l_{in}, W^l_{out} は、多層化するに従ってパラメータ数が増加するため、大きな計算資源を必要とする主要な要因になりうる．

この問題を解消するために、本研究では層間で重み W^l_{in}, W^l_{out} の共有を行う．具体的には、任意の l 層の重み $W^l_{in}, W^{l,fwd}_{out}, W^{l,bwd}_{out}$ に対して、l に依らない共通の重み $W_{in}, W^{fwd}_{out}, W^{bwd}_{out}$ を割り当てる．この時、層間で重みを共有することで層を重ねる方向に対して、重み W^l_{in}, W^l_{out} は再帰的重みとなるが、入力系列に対して系列方向に再帰的ではないため並列処理性を失うことはない．

Encoder Model	Num of Params	Compressibility
BiLSTM[11]	788,480	1.000
Selective BiLSTM[16]	1,313,280	1.665
LSTM	394,240	0.500
FFNN	493,568	0.625
ACNN(proposed)	331,776	0.420

表 2: 各 Encoder モデルのパラメータ数と BiLSTM に対するパラメータ圧縮比率．ACNN は実験で用いる 3 層モデルのパラメータ数を表す．

4 実験

長文要約データセット用いて提案手法の有効性を検証する．実験には、CNN/DailyMail データセットを用いる．データセット構成は、訓練データが 287,226 ペア、開発データが 11,490 ペア、テストデータが 11,490 ペアとなっている．また、語彙の大きさを頻出 50,000 語として実験を行う．モデル学習時の入出力については、最大入力長と最大出力長をそれぞれ 400 語、100 語として学習を行う．一方で、評価時には最大出力長を 120 語として生成を行い、ビーム幅を 4 とするビームサーチから予測系列を得る．評価は、ROUGE 値と METEOR 値による精度比較[5]を行う．なお、データセットと実験設定は、先行研究 [11] と同様のものである．

4.1 実装詳細

提案モデルの設定について述べる．提案モデルは、3 層の ACNN に対して各層の入出力に residual 接続を施したモデルを用いた．入力部の単語埋め込み表現ベクトルとエンコード特徴量の次元は、それぞれ 128、256 とした．また、コンテクスト窓の大きさと数は全ての層で大きさを 20、窓数を 1 とした．また、これらは前方向と後方向で同様である．ミニバッチサイズは 16 とし、初期学習率は 0.15 とした．

モデル訓練は、clip-gradient の大きさを 2 として学習時の最適化法に Adagrad を用いて数式 (2) を最小とするように学習を行った．提案モデルのパラメータの初期化には、重み行列とバイアス項にそれぞれ xavier 初期化 [7] とゼロ初期化を行った．訓練は、attention 機構のみを用いた場合は 250000 iters 程度、pointer-generater 機構を用いた場合は 30000 iters 程度を行なった．また、coverage 機構による検証は、pointer-generator 機構を用いた場合の最高精度のモデルに対して、さらに 5000 iters 程度の再学習を行う．

実装には python(2.7 系) と Tensorflow(ver.1.2) [1] を使用した．実験システムは、エンコーダのみを比較し、その他の構成部は各比較モデルの実験で共通とした．また、実験環境は理研 AIP の深層学習用大型計算機 RAIDEN[6] 上で行った．

4.2 比較手法

比較手法のエンコーダには、先行研究 [11] で最高精度の BiLSTM といくつかのモデルで比較を行う．比較するエンコーダには、再帰型ニューラルネット

[5]紙面の都合上、各エンコーダモデルによる実際の生成要約文と正解要約文の比較による定性評価は割愛させて頂いた．

[6]構成は、CPU : Intel Xeon E5-2698 v4 / GPU : Tesla P100. (2018.01 時点) 本実験では、GPU を 1 枚利用した．

Encoder Model	ROUGE			METEOR	
	1	2	L	exact match	full (+ stem/syn/para)
attention only					
BiLSTM[11]	31.33	11.81	28.83	12.03	13.20
BiLSTM (re-exam)	33.24	13.11	29.68	13.41	14.70
ACNN	**33.65**	**13.41**	**30.38**	**13.65**	**14.93**
+ pointer-generator					
BiLSTM[11]	36.44	15.66	**33.42**	15.35	16.65
BiLSTM (re-exam)	36.67	15.55	32.13	15.07	16.33
ACNN	**36.78**	**15.68**	32.29	**15.70**	**17.02**
+ pointer-generator + coverage					
BiLSTM[11]	39.53	17.28	**36.38**	17.32	18.72
BiLSTM (re-exam)	39.35	17.09	34.83	17.52	18.93
ACNN	**39.57**	**17.40**	35.07	**18.89**	**20.48**

表 3: pointer-generator/coverage 機構を用いた場合の ROUGE 値 (F-score) と METEOR 値による評価.

ワークに Selective-BiLSTM[16][7], LSTM の 2 つを用いる．また、非再帰型ニューラルネットワークに feed forward neural network(FFNN) をエンコーダに用いる．FFNN は、活性化関数に $tanh$ 関数を用いた linear layer を 8 層重ね、各層の入出力を residual 接続したモデルを構成した．エンコーダは全て単語埋め込み表現ベクトルの次元を 128、隠れ層の次元を 256 として設定した．また、非再帰型ニューラルネットに対しては、単語埋め込み表現に加え、単語列の位置情報を与える position embedding[10, 15] を用いた．position embedding の次元は 128 とした．各々のパラメータは、初期値に $[-0.01, 0.01]$ の一様分布に従ってサンプリングした値を設定した．

4.3 実験結果

最初に、attention 機構のみで実験を行った結果を表 1 に示す．ROUGE は F 値を示す．提案モデルの ACNN が先行研究の BiLSTM の再現値よりも、ROUGE-1、ROUGE-2、ROUGE-L でそれぞれ 0.41、0.30、0.7 ポイント上回っている．また、表 2 から、ACNN が BiLSTM の 42%ほどのパラメータで構成されており、先行研究よりも小さなモデルで高い精度を実現していることが分かる．

次に、pointer-generator 機構と coverage 機構を用いた実験の結果を表 3 に示す．pointer-generator 機構を用いた実験では、ACNN が先行研究の BiLSTM よりも ROUGE-1、ROUGE-2 でそれぞれ 034.、0.02 ポイント上回っている．また、METEOR による評価では、exact mode と full mode でそれぞれ 0.35、0.37 ポイント上回っている．coverage 機構を用いた実験では、METEOR 評価で exact match mode と full mode 共に 1.0 ポイントを超える高い精度を示している．このことから、ACNN は並列処理性能を備えているだけでなく系列性を考慮できることから、BiLSTM と同等のエンコード性能を期待できる．

5 おわりに

本稿では、隣接畳み込みニューラルネットワーク (ACNN) を提案した．ACNN は、入力系列に対して系列情報を考慮しつつ系列に対して並列処理が可能である．このため、従来の再帰型ニューラルネットワークを用いて長い入力系列のエンコード時を行う場合に生じる勾配爆発・消失の問題を回避しつつ並列エンコードが可能である．また、実験では、長文要約データセットを用いて先行研究の BiLSTM エンコーダと同程度以上の精度を達成した．今後の研究として、複数文書を対象とした生成要約への提案モデルの適用を考えている．

[7] 論文 [16] に基づいて、python で再実装したものを用いた．

参考文献

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

[2] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014.

[3] S. Banerjee and A. Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, volume 29, pages 65–72, 2005.

[4] S. Chopra, M. Auli, and A. M. Rush. Abstractive sentence summarization with attentive recurrent neural networks. In *NAACL: Human Language Technologies*, pages 93–98, 2016.

[5] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier. Language modeling with gated convolutional networks. In *ICML 2017*, pages 933–941, 2017.

[6] J. Gehring, M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin. Convolutional sequence to sequence learning. In *ICML 2017*, pages 1243–1252, 2017.

[7] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010.

[8] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*, volume 8. Barcelona, Spain, 2004.

[9] R. Nallapati, B. Zhou, C. N. dos Santos, Ç. Gülçehre, and B. Xiang. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *CoNLL 2016*, pages 280–290, 2016.

[10] A. M. Rush, S. Chopra, and J. Weston. A neural attention model for abstractive sentence summarization. In *EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 379–389, 2015.

[11] A. See, P. J. Liu, and C. D. Manning. Get to the point: Summarization with pointer-generator networks. In *ACL 2017*, pages 1073–1083, 2017.

[12] S. Semeniuta, A. Severyn, and E. Barth. A hybrid convolutional variational autoencoder for text generation. In *EMNLP 2017*, pages 627–637, 2017.

[13] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014.

[14] J. Suzuki and M. Nagata. Cutting-off redundant repeating generations for neural abstractive summarization. In *EACL 2017*, pages 291–297, 2017.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *NIPS 2017*, pages 6000–6010, 2017.

[16] Q. Zhou, N. Yang, F. Wei, and M. Zhou. Selective encoding for abstractive sentence summarization. In *ACL 2017*, pages 1095–1104, 2017.