出力長制御を考慮した見出し生成モデルのための大規模コーパス

人見 雄太[1] 田口 雄哉[1] 田森 秀明[1] 菊田 洸[2] 西鳥羽 二郎[2]
岡崎 直観[4] 乾 健太郎[5,6] 奥村 学[4]
株式会社朝日新聞社[1] 株式会社レトリバ[2] 東京工業大学[3]
東北大学[4] 理化学研究所 AIP センター[5]
{hitomi-y1, taguchi-y2, tamori-h}@asahi.com, kikutakou@gmail.com,
jiro.nishitoba@retrieva.jp, okazaki@c.titech.ac.jp,
inui@ecei.tohoku.ac.jp, oku@pi.titech.ac.jp

1 はじめに

ニュースメディアはより多くの読者に記事を配信し，閲覧してもらうため，同一の記事を異なるメディア，デバイス，アプリケーションに向けて配信している．配信先の画面サイズ，レイアウト，デザインに応じて，見出しの文字数に厳密な制約が課せられている．例えば表 1 では，1 つの記事に対して，紙面用の見出しに加え，配信先に応じて 10，13，26 文字以下の見出しが付けられている．近年，Encoder-Decoder モデルを利用して記事本文から見出しを生成する研究 [1, 2] が盛んに行われているが，見出し生成を記事配信の現場に適用するには，所望の文字数以下で見出しを生成する技術が必要である．

Encoder-Decoder における出力長制御に最初に着目したのは Rush ら [1] である．この研究では，必要なトークン数に達するまで終端記号 (EOS) のスコアを強制的に $-\infty$ にすることで，出力長を制御した．菊池ら [3] と Fan ら [4] は，Sequence-to-Sequence (Seq2Seq) [5] の枠組みで，出力長を制御する手法を提案した．彼らの研究では，30，50，75 文字を指定し生成した要約を，75 文字を目標に作られた要約 (DUC 2004) [1]で評価した．つまり，システム出力と参照要約との間で圧縮率が異なる状況下で，強制的に評価していることになる．このため，短い要約だからこそ用いるべきキーワードが含まれていなくても，長い参照要約に含まれる単語のいずれかを出力すれば評価スコアが高くなってしまうという問題があった．

本稿では，出力長を考慮した見出し生成モデルを評価するためのコーパス JApanese MUlti-Length Headline Corpus (JAMUL)[2]と，日本語の見出し生成のための学習コーパス Japanese News Corpus (JNC)[3]を提案する．また，本コーパスを用いて従来手法を評価し，見出し生成の手法，およびその評価法について議論する．なお各コーパスの入手方法は，脚注のウェブサイトを参照されたい[2,3]．

記事
人工衛星から金属球を打ち出して流れ星のように見せる技術を開発するベンチャー企業「ALE」(本社・東京、岡島礼奈社長) は、2019 年初夏に広島・瀬戸内地域で「人工流れ星」の実験を行うと発表した。...
紙面見出し
流れ星、降らせます 19 年、金属球使って実験
デジタル見出し
10 文字　人工流れ星、初実験へ 13 文字　人工流れ星、19 年初夏実験 26 文字　人工流れ星輝くか、19 年初夏に実験 広島・瀬戸内地域

表 1: 複数の長さの見出しが付与された例

2 JNC と JAMUL

2.1 見出し付与のフロー

本節では新聞社が見出しを付与するフローの一例を説明する．記者が書いた記事は，新聞紙面を編集する部署に送信される．まず，この部署によって，新聞紙面用の見出し（紙面見出し）が付与される．

次に，紙面用に配信された記事の中から，デジタル版記事として配信するものを別の部署が選び，各記事に対して長さの異なる複数の見出し（デジタル見出し）を付与する．電光掲示板や音声メディア向けの見出しは，ひと目で記事の内容を理解してもらうために 10 文字以下で書かれる．小さい画面を持つガラパゴス携帯やニュースサイトのアクセスランキングに表示するための見出しは，デバイスやニュースサイトのレイアウトに合わせて 13 文字以下で書かれる．主に PC 向けニュースサイト上で表示し，記事本文にリンクするための見出しは，26 文字以下で書かれる．このように実サービスにおける見出しは，デバイスやレイアウトによって文字数の上限が異なる．

[1] https://duc.nist.gov/DUC2004/
[2] http://www.asahi.com/shimbun/medialab/JAMUL/ (無償)
[3] http://www.asahi.com/shimbun/medialab/JNC/ (有償)

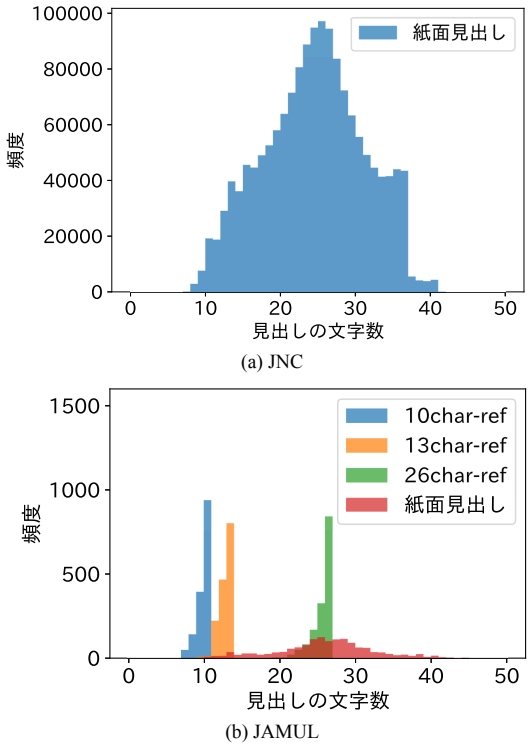

図 1: JNC と JAMUL が収録する見出しの長さ分布

JNC と JAMUL はプロの編集者が上記のフローにおいて付与した見出しを収集・構築したコーパスである．

2.2 JNC

JNC は，10 年間（2007〜2016 年）の記事[4]と紙面見出しのペア 1,831,812 件を収録したコーパスである．図 1(a) は，JNC に収録されている見出しの長さのヒストグラムである．JNC に収録されているのは紙面見出しのみだが，様々な長さの見出しを含んでいることが分かる．これは，紙面の掲載スペースによって見出しの長さは大幅に変わるためである．このように，JNC には様々な長さの見出しが含まれるため，一般的な見出し生成モデルだけでなく，出力長を考慮した見出し生成モデルの学習データとしても適切である．

2.3 JAMUL

JAMUL には，2017 年 9 月から 2018 年 3 月の間に配信された記事[5]と紙面見出し，および 10，13，26 文字の各デジタル見出しが 1,524 件収録されている．プロの編集者によって各記事に対して複数の長さの見出しが付与されている点が，従来のコーパスとは大きく異なる点である．以下，JAMUL に収録されているデジタル見出しをそれぞれ 10char-ref，13char-ref，26char-ref と呼ぶ．

2.4 紙面見出しとデジタル見出しの比較

学習用の JNC は紙面見出し，評価用の JAMUL はデジタル見出しと紙面見出しを収録している．見出しの長さの制約が厳密であるデジタル見出しを評価に用いる場合，学習データと評価データの見出しの掲載先が異なることになる．そこで，それぞれの見出しがどの程度似ているのか，特に，デジタル見出しが紙面見出しに対してどの程度似ているのかを検証した．具体的には紙面見出しを「システム出力」，10char-ref，13char-ref および 26char-ref のそれぞれを「参照要約」とみなし，ユニグラム[6]の適合率と再現率を測定した（表 2）．再現率が 60%前後であるので，デジタル見出しで用いられる語彙の半数以上は紙面見出しに由来する一方で，デジタル見出しの約 4 割の単語は，紙面見出しに表れないことを示唆している．

2.5 異なる長さのデジタル見出し間の比較

JAMUL には 1 つの記事に異なる長さのデジタル見出しが付与されている．そこで，短い見出しは長い見出しの一部を取り出すだけで作れるのか，という疑問を検証してみたい．具体的には，26char-ref をそのまま，あるいは先頭および末尾から 10 文字，13 文字をトリミングしたものを「システム出力」，10char-ref および 13char-ref を「参照要約」とみなしてユニグラムの適合率および再現率を測定し，結果を表 3 に示した．

表 3 の 1，2 行目は 26char-ref をそのままシステム出力として評価した場合である．これらは，26char-ref を適切に圧縮して 10char-ref，13char-ref を生成した場合の再現率の上限を示す．いずれも高い再現率を示しており，26char-ref は 10char-ref および 13char-ref で使われる単語の多くを被覆することが分かった．

3，4 行目は 26char-ref の先頭 10 文字，13 文字をシステム出力，10char-ref，13char-ref を参照要約としたときの適合率および再現率を示す．先頭 13 文字と 13char-ref の比較では，おおむね高い適合率・再現率を示すことから，26char-ref の先頭 13 文字で 13char-ref の代用がある程度可能であることが示唆される．一方，先頭 10 文字と 10char-ref との比較では再現率が極端に低下する．したがって，26char-ref と 10char-ref の構成はに大きな差があることがわかる．また，5，6 行目は 26char-ref の末尾 10 文字，13 文字との比較である．再現率がかなり低下するが，26char-ref の末尾の

[4]記事は先頭（リード）から 3 文のみを収録．
[5]JNC と揃えるため，リードの 3 文のみを収録している．
[6]本研究では，単語分割には MeCab[6] と IPA 辞書を用いた．

システム出力	参照要約	適合率	再現率
紙面見出し	10char-ref	24.66	64.11
紙面見出し	13char-ref	33.24	66.30
紙面見出し	26char-ref	56.36	55.75

表 2: 紙面見出しとデジタル見出し間のユニグラムでの適合率および再現率

システム出力	参照要約	適合率	再現率
26char-ref	10char-ref	28.77	78.55
26char-ref	13char-ref	42.53	88.75
26char-ref の先頭 10 文字	10char-ref	38.40	41.65
26char-ref の先頭 13 文字	13char-ref	60.31	65.58
26char-ref の末尾 10 文字	10char-ref	14.55	17.05
26char-ref の末尾 13 文字	13char-ref	23.13	26.56

表 3: JAMUL に収録された 26char-ref と 10char-ref, 13char-ref の比較

部分にも短い見出しを構成する単語がある程度含まれることを示している．

3 JAMUL による従来手法の比較

本節では，出力長を考慮した見出し生成の実験を報告する．従来手法とその組み合わせによる見出し生成モデルを大規模学習データ（JNC）で学習し，記事配信に実際に用いられたデータセット（JAMUL）で性能を評価し，その出力結果を分析する．

3.1 出力長制御を考慮した見出し生成

この実験では，4 つの出力長制御手法を検証する．最初の 2 つは，菊池ら [3] が提案した *LenEmb* と *LenInit* で，どちらも Seq2Seq を拡張した手法である．*LenEmb* は，所望の出力長の情報を長さ埋め込みベクトルで表現し，Decoder に入力するもので，*LenInit* は Decoder のメモリセルの初期値に所望の長さを表すスカラー値を掛け合わせる手法である．3 つ目は，Fan ら [4] の Convolutional Seq2Seq (ConvS2S) [7] にスペシャルトークン (*SP-token*) を用いる手法である[7]．4 つ目は，*LC* [8] と呼ばれる手法である．ConvS2S を拡張した手法で，Residual Connection [9] の初期値に所望のトークン数を掛け合わせるものである[8]．

以上 4 つの出力長制御に加え，*SP-token* については Seq2Seq や Transformer [10] との組み合わせについても実験し，最終的に本実験では表 4 に示す 6 つのモデルを比較する．

[7]元論文ではおおよその長さを表現したトークンを用いていたが，本実験では長さ毎にトークンを用意した．

[8]本実験では，所望する文字数を掛けた．

3.2 実験設定

表 4 に示す 6 つのモデルを JNC で学習した．フィルター処理[9]により，1,568,360 件の紙面見出しと記事のペアを取得した．更にそのうちの 1%を検証データとして利用した．語彙の構築に Byte Pair Encoding (BPE) [11] を用い，その Merge Operation を 8000 と定め，記事本文と見出しを結合したコーパスで語彙を構築した．学習時には，それぞれの参照見出しの長さを出力長として指定した．推論時には，JAMUL の各上限の長さである 10，13，26 文字を出力長に指定した．JAMUL も同様にフィルター処理[9]をした上で、1,181 件で評価した。評価は再現率ベースの ROUGE-1，ROUGE-2，ROUGE-L を用いた．なお，出力された見出しが指定出力長を超える場合は，先頭からその指定出力長までで切り落として評価した．

Seq2Seq の実装として OpenNMT[10] を，ConvS2S および Transformer の実装として fairseq[11] を用いた．*LenEmb*，*LenInit*，*LC* もこれらの実装をベースにしている．トークン，長さ埋め込み，隠れ層は 512 次元，ビーム幅は 5 とした．ConvS2S は Nesterov's Accelerated Gradient Method (NAG) で最適化を行い，その Momentum を 0.99 とした．Transformer では，Attention Head 数を 8，順伝搬層を 2048 次元，Adam の β_2 を 0.98，Warming up を 4000 ステップ，Label smoothing の ϵ を 0.1 とそれぞれ設定した．表 5 に，その他のパラメータを示した．

3.3 JAMUL による各モデルの評価

表 4 に，JAMUL で測定した ROUGE スコアを示す．本実験においては *SP-token* を用いたモデルが概ね良好な性能を示しており，特に Transformer + *SP-token* が最も高い性能を示した．

従来手法における DUC 2004 での評価のように，単一の長さの参照見出しのみで評価した場合の影響を調査した．表 6 は，10 文字または 13 文字の出力を 26char-ref で評価した場合の結果である．この場合でも Transformer + *SP-token* が最も高いスコアを示しているが，ROUGE の絶対値が小さくなり，それに伴い性能差も小さくなっている．また，出力長が 13 文字の評価では Seq2Seq + *SP-token* よりも Seq2Seq + *LenInit* のスコアが大きいなど，モデル間の ROUGE スコアの順位の入れ替わりが見られる．表 7 に，10 文字を出力長として指定した場合のシステム出力を，10char-ref および 26char-ref を参照要約として評価したときに，スコアの順位が入れ替わる例を示す．10char-ref で

[9]https://github.com/asahi-research/Gingo

[10]https://github.com/OpenNMT/OpenNMT-py

[11]https://github.com/pytorch/fairseq

	10 文字			13 文字			26 文字		
モデル	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L
(1) Seq2Seq + *LenEmb*	34.66	15.29	33.56	40.66	19.40	38.23	44.82	20.75	36.62
(2) Seq2Seq + *LenInit*	36.50	16.75	35.54	41.40	19.49	38.83	46.77	22.06	38.29
(3) Seq2Seq + *SP-token*	38.09	17.43	36.67	42.51	19.79	39.76	47.33	22.12	38.59
(4) ConvS2S + *SP-token*	38.90	17.84	37.53	43.32	20.35	40.31	47.10	21.51	37.86
(5) ConvS2S + *LC*	37.71	16.89	36.50	42.60	20.11	39.97	45.76	21.93	37.91
(6) Transformer + *SP-token*	42.85	19.84	41.40	46.92	22.85	44.09	51.57	24.52	41.05

表 4: それぞれの手法の ROUGE．参照要約は JAMUL，出力長は 10，13，26 文字とした．

	Seq2Seq	ConvS2S	Transformer
Num of Layer	2	8	6
Dropout Rate	0.3	0.1	0.3
Grad Clipping	[-5.0, 5.0]	[-0.1, 0.1]	-
Learning Rate	0.001	0.2	0.001
Optimizer	Adam	NAG	Adam

表 5: 実験で使用した各パラメータ

	10 文字			13 文字		
モデル	R-1	R-2	R-L	R-1	R-2	R-L
(1)	21.22	8.56	19.29	27.42	12.20	24.23
(2)	21.72	9.03	19.70	28.34	12.83	25.01
(3)	22.02	9.08	20.03	28.26	12.09	24.73
(4)	22.50	9.25	20.41	29.23	12.82	25.58
(5)	22.64	9.27	20.57	28.94	12.53	25.21
(6)	24.36	10.32	21.97	31.22	13.99	27.25

表 6: 26char-ref を参照要約として，10 文字，13 文字を出力長として指定したシステム出力の評価（ROUGE）．モデルの (1)∼(6) は表 4 のモデル (1)∼(6) を示す．

評価した場合は (6) のモデルの出力が最も高いスコアを得ているが，26char-ref で評価した場合は (1)∼(4) のモデルからの出力が最も高くなっている。

以上の分析だけでは，出力したい長さに応じた正解セットを用いるべきなのか，異なる長さの見出しで代用することが可能なのか，結論を出すことはできない．そもそも，異なる長さの見出しは異なる用途を想定して書かれているため，紙面の見出しのみを学習データとするのではなく，用途に応じた分野適用などが今後の展開として有望かもしれない．

4 おわりに

本稿では，出力長を制御できる見出し生成モデルを学習・評価するための 2 つの大規模コーパスを提案した．また，本稿で提案したコーパスを用いて異なる長さの参照見出しの比較，および従来手法の比較を行い，見出しの出力長制御の研究における課題を議論した．

参照要約			
10	米娯楽大手が売却交渉		
26	21 世紀フォックス、ディズニーに事業売却交渉 米報道		
システム出力（出力長 : 10 文字）			
モデル	見出し	10	26
(1)	21 世紀フォックス	0.00	25.00
(2)	フォックス、自社売却	16.67	25.00
(3)	フォックス、事業協議	0.00	25.00
(4)	米メディア、売却協議	33.33	25.00
(5)	米メディア大手売却へ	50.00	16.67
(6)	米娯楽大手が売却協議	83.33	16.67

表 7: 10char-ref および 26char-ref を用いた、システム出力の評価（ROUGE-1）．表中の 10 および 26 は 10char-ref，26char-ref を示す．

参考文献

[1] Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. In *EMNLP*, pp. 379–389, September 2015.

[2] Jun Suzuki and Masaaki Nagata. Cutting-off redundant repeating generations for neural abstractive summarization. In *EACL*, pp. 291–297, April 2017.

[3] Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. Controlling output length in neural encoder-decoders. In *EMNLP*, pp. 1328–1338, 2016.

[4] Angela Fan, David Grangier, and Michael Auli. Controllable abstractive summarization. In *WNMT*, pp. 45–54, 2018.

[5] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *NIPS*, pp. 3104–3112, 2014.

[6] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying conditional random fields to japanese morphological analysis. In Dekang Lin and Dekai Wu, editors, *EMNLP*, pp. 230–237, July 2004.

[7] Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. Convolutional sequence to sequence learning. In *ICML*, pp. 1243–1252, 2017.

[8] Yizhu Liu, Zhiyi Luo, and Kenny Zhu. Controlling length in abstractive summarization using a convolutional neural network. In *EMNLP*, pp. 4110–4119, 2018.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pp. 770–778, 2016.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pp. 6000–6010, 2017.

[11] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *ACL*, pp. 1715–1725, August 2016.