Vol.2019-NL-241 No.28
2019/8/30

生成型文要約のための抽出性に着目したデータ選択

長谷川 駿[1,a)]　上垣外 英剛[1,b)]　奥村 学[1,c)]

概要：生成型文要約は必ずしも原文の語句を抽出することなく，入力に対して極めて柔軟な要約文を生成することが可能である．しかし，我々の事前調査で，最高精度に近い性能を達成している文要約器の出力では，原文から借用した単語が生成文の約 8 割弱を占めていることが判明している．一方で，その要約器の学習に用いた訓練データでは，参照要約文において原文から借用された単語は約 6 割弱にとどまっている．我々は，これらの調査結果における実際の生成文と訓練データの抽出率の乖離から，既存の生成型文要約器が抽出的な要約を得意としており，抽出率の低いデータ対が学習時のノイズとなっているという仮定を置いた．本研究ではこの仮定に基づき，訓練データから抽出率の低いデータ対を除去する，簡易で効果的なデータ選択手法を提案する．実験の結果，提案手法を用いた場合，3 つの種類の文要約器において全データで学習した場合の半分のデータ量・学習時間で同等の要約性能を達成できることを確認した．また，訓練データの抽出性・生成性を変化させて学習・比較を行うことで，それらの訓練データの性質が文要約器に与える影響の分析も行った．

キーワード：要約，文要約，生成型要約，機械翻訳，深層学習，データ選択

1. はじめに

文要約とは，任意の文をその大意を保持しつつ短くするタスクである．これまで文要約の研究は盛んに行われてきたが，その主流は文圧縮，つまり文法制約を満たすように原文中の単語を抽出することで文を要約する手法であった．文圧縮はいわゆる抽出型要約の 1 種であり，重要語の抽出，文法性の担保がある程度容易である一方，フレーズを短く言い換え構文構造を変化させるなど，我々人間が文の要約を “生成”する際に行うような操作を扱えない．そのため，表 1 のような “抽象的な” 要約を出力可能な文要約技術が渇望されていた．

表 1　文要約の例．

原文 1	私は巨人で選手としてプレイしています
抽出的な要約	私は巨人でプレイしています
抽象的な要約	私は巨人の選手です
原文 2	アメリカ人とドイツ人が観光のため日本を訪れた
抽出的な要約	アメリカ人とドイツ人が日本を訪れた
抽象的な要約	外国人 2 人が訪日した

近年，ニューラルネットによる系列変換モデルの発展により，大規模な訓練データさえ用意できれば文圧縮よりもより人間の生成に近い文要約，つまり生成型要約が可能となった [1]．文献 [1] 以降，系列変換モデルの様々な改良が提案され，現在に至るまで文要約手法の主流は抽出型から生成型へと移っていったと言える．

しかし近年の研究で，生成型文要約には様々な問題があることがわかってきた [2]．たとえば，冗長な単語の出力，不要な単語が出力される，事実ではない情報が含まれる，重要な情報が足りていない等である．こうした問題を解決

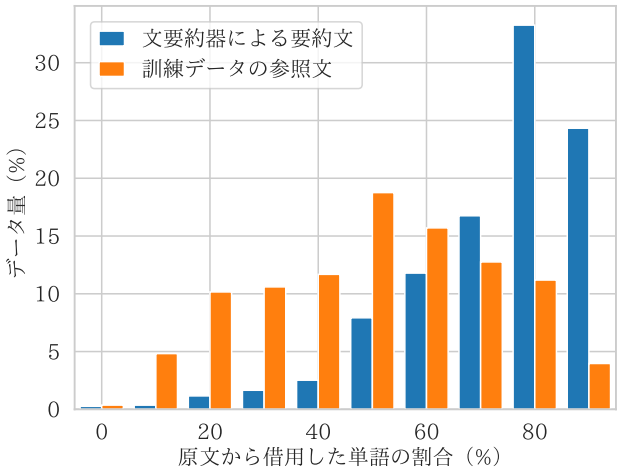

図 1　原文から単語を借用した割合の分布（マクロ平均）．要約モデルは後述する COPY．

1　東京工業大学
a)　hasegawa.s@lr.pi.titech.ac.jp
b)　kamigaito@lr.pi.titech.ac.jp
c)　oku@pi.titech.ac.jp

するため，原文の語句が要約に多く含まれるよう，原文の語句を再利用する系列変換モデルが提案され，性能向上が報告されている [3]．実際，図 1 に示す我々の事前調査によると，現在，最高性能と思われる要約システムによる要約文の 74.3%で，原文から借用した単語が 70%以上を占めている．つまり，現状の生成型要約手法は原文の語句を再利用した，抽出率の高い要約を生成しており，それが性能向上につながっていることを示唆している．

一方，要約システムの学習に利用した訓練データである，原文・要約 (以降，参照文) のペア集合には，原文から借用した単語の割合が 50%未満である参照文が 37.6%も含まれている．このような低い抽出率の要約文が要約システムから生成される割合は 6.6%と少なく，訓練データと要約システムの抽出率には大きな乖離がある．よって，要約システムが抽出率の高い要約を生成することで高い性能を発揮しているのであれば，原文と抽出率の低い参照文のペアは要約モデルの学習においてノイズになると考えられる．

そこで本研究では，簡易で効果的なデータ選択手法を提案し，高い抽出率の参照文のみで構成された訓練データを用いることで，少ないデータでも現状の性能とほぼ変わらない要約システムが実現できることを示す．さらに，訓練データの抽出性が文要約器に与える影響を調査するため，訓練データの抽出性・生成性を変化させて要約システムの出力に対する分析を行う．

2. 関連研究

生成型文要約が盛んに研究されるきっかけとなった Rush ら [1] の一番の貢献点は，容易に入手可能な大規模訓練データを用意したことである．彼らは新聞記事の一文目とタイトルを原文・要約文のペアとみなし，ノイズを取り除くためのフィルタを通過したデータ対を訓練データとした．しかし，本研究で着目する抽出率のような原文・要約文の関係性は「3 文字以上の単語が 1 単語は共通していなければならない」というフィルタでしか言及していない．

データ選択手法は，生成型文要約と同じ文生成タスクである機械翻訳や対話で研究が行われている．機械翻訳では，WEB 資源から集めた誤訳等のノイズを多く含む大規模対訳コーパスから，クロスエントロピーや単語アライメントの確率を用いてノイズとなるデータを除去する手法が提案され，ニューラル翻訳モデルの性能向上を達成している [4], [5]．また，翻訳モデルを使用するドメインと訓練データのドメインが離れている場合に，言語モデルを用いて使用するドメインに近いデータを訓練データとして選ぶ手法も提案されている [6]．さらに，人工的に様々なノイズを訓練データに加えて翻訳モデルを学習することで，ノイズがモデルに与える影響を調査する研究も行われている [7], [8]．近年では機械翻訳におけるデータ選択に関するシェアードタスクも存在する [9]．

対話でも，訓練データ中の応答ペアの関連性を定量化し，データ選択を行う手法が提案されている [10], [11]．

一方要約では，入力と出力の含意関係に着目したデータ選択手法が提案されている [12]．松丸らは新聞記事最初の 3 文からヘッドラインを生成するタスクを対象とし，訓練データ中の入力と出力の含意関係を検出する識別器を作成しデータ選択を行うが，システム出力の原文に対する忠実性（faithfulness）を指向した研究であり，本研究とは目指している方向が異なっている．本研究で扱う生成型文要約においてデータ選択を行う研究はこれまで行われていない．

また，事前調査を行なった要約システムによる要約の抽出性は，近年注目を集めている．特に生成型文書要約においては，要約システムによる要約の抽出性が極めて高いことが指摘されており [13], [14]，その後，モデルの構造を変えることで要約システムによる要約の抽出性をあげたり [13], [14]，逆に生成性をあげたりする [15] 研究がそれぞれ行われている．ただし，生成性をあげた要約システムによる要約においても，依然として訓練データの抽出性より高い抽出性となっている．また，文書要約用コーパスの参照文書における原文から借用した語句の割合についても言及がなされている [16]．しかし，訓練データの抽出性が要約システムにどのような影響を与えるかの分析は行われていない．

3. 抽出率に基づくデータ選択

本稿では，訓練データ中の任意の参照文 t のその原文 s に対する抽出率を ROUGE-1(再現率) を用いて以下の式で定義する．

$$\mathrm{extr}(t,s) = \mathrm{ROUGE{-}1}(t,s) \tag{1}$$

抽出率が高いことは，参照要約が原文中の単語を再利用して生成されたことを示し，低いことは参照要約が原文中の単語を再利用せず新しい単語で生成されたことを示す．ただし，単語の一致は表層の一致 (以降，コピー) だけでなくステムの一致 (以降，ステムコピー) も考慮する．これは，現状の系列変換モデルが語形変化程度ならば容易に対応可能であろうと考えたからである．よって，抽出率は ”`-n 1 -m`” オプションを用いて `ROUGE-1.5.5.pl` で計算したスコアに一致する．

抽出率が閾値未満のデータ対を訓練データから除去することで，抽出率の高いデータ対のみで構成された学習データを得る．閾値を minR1 とすると，得られる学習データ $extrD_{minR1}$ は以下のように表せる．

$$extrD_{minR1}=\{(t,s)\in D|\mathrm{ROUGE{-}1}(t,s)\geq minR1\} \tag{2}$$

ここで，$\mathrm{ROUGE{-}1}(t,s)$ は参照文 t に対する原文 s の ROUGE-1（再現率），D は全データ対である．

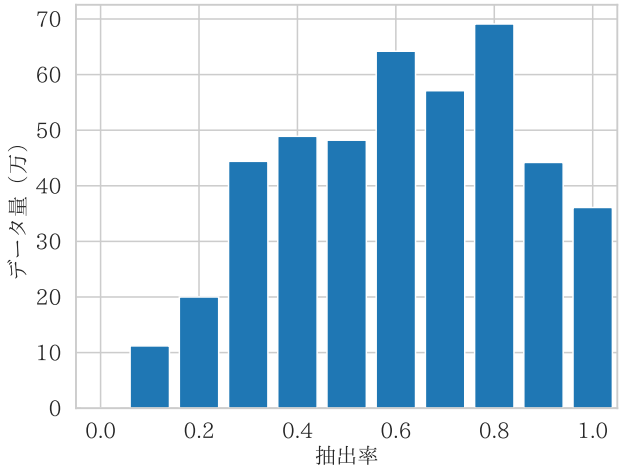

図 2 訓練データにおける抽出率の分布.

4. 実験

提案法の有効性を確認するため，作成した学習データで 3 つの文要約器を学習し比較実験を行う．最初に実験に用いるデータセットと文要約器について述べ，実験結果を報告する．

4.1 データセット

一般的に系列変換モデルの学習には大量のデータがあった方がより良い性能が出せると言われている．よって，Rush ら [1] が用いた Gigaword コーパス [17] から作成される訓練データに加え，Newsroom[16], New York Times[18] コーパスから作成したデータ対も訓練データとして利用した．各コーパスは公開されているデータ生成・前処理スクリプト[*1]により処理を行なった．最終的に 440 万文対の訓練データ，1 万文対の評価データ，1 万文対の開発データを得た[*2]．ただし，Rush らの処理とは異なり低頻度単語を未知語タグには置き換えていない．訓練データにおける抽出率の分布を図 2 に示す．

訓練データに対して提案法であるデータ選択手法を適用し，学習データを変化させ要約モデルの訓練，評価を行う．提案法の閾値 minR1 には 0.1 刻みの 10 段階（0 〜 0.9）を用いた．ただし，閾値 0 はデータ選択を行わないことを意味することに注意されたい．各閾値における学習データのデータ量と平均抽出率を表 2 に示す．

4.2 ニューラル生成型文要約器

本実験では 3 つの文要約モデルを学習する．1 つ目は翻訳をはじめとした言語生成タスクで広くベースラインとして用いられている注意機構付き両方向系列変換モデル（ATTN），2 つ目は未知語も含め原文から単語を借用することのできるコピー機構 [13] を ATTN に付属したモデル（COPY），3 つ目は情報の取捨選択ができる selective gate network[19] を ATTN に付属したモデル（SELE）である．

ATTN と COPY の実装には OpenNMT[20] を用いた．各実験設定は OpenNMT の学習済みモデルに従っている．SELE には chainer[21] による実装[*3]をベースに，selective gate network を付属したプログラムを使用している．SELE の主な実験設定は Zhou ら [19] に従っている．ただし，ATTN や COPY と実験設定を揃えるため，SELE に用いる RNN のユニットは LSTM[22] に変更し，Luong らのデコーダ [23] を使用した．また，学習時間を減らすため，SELE では頻度 10 未満の単語を未知語として扱い，バッチサイズを 256 に変更している．

最終的な学習済みモデルには，学習の各エポック終了時に計測する開発データにおける ROUGE-2（F 値）が最良となるモデルを選択する．

表 2 各閾値 minR1 による学習データのデータ量と平均抽出率．抽出率は参照文をターゲットとする原文の ROUGE-1（再現率）．

閾値	ALL	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
文対数 (万)	445	443	426	391	354	315	244	178	115	47
除去率 (%)	0	0.4	4.3	12.1	20.4	29.2	45.1	60.0	74.2	89.4
平均抽出率	0.61	0.61	0.63	0.66	0.70	0.73	0.79	0.84	0.90	0.98

4.3 比較実験結果

図 3 に閾値を変化させた場合の ROUGE-2 スコアの変化を示す．なお，系列変換モデルの初期値依存問題を軽減するため，異なる初期値で学習を 5 回行いそれぞれ得た ROUGE スコアの平均値を最終的な評価スコアとした．

図 3 中の実線は提案法，破線は提案法と同じデータ数を訓練データからランダムに選んだ結果を示す．なお，左端 0.0(ALL) は，全訓練データを用いた場合，すなわちデータ選択を行わない場合を示す．図より，ほぼ全ての閾値・モデルで $randD_{minR1}$ より $extrD_{minR1}$ が良いスコアを得ていることがわかる．これは，提案法のデータ選択の基準に意味があることを示唆している． また，全モデルで閾値 0.4 の場合に最も良いスコアとなっており，20%のデータ対が除去されているにも関わらず（表 2 参照）全訓練データで学習した場合よりも約 0.2 ポイントほど高いスコアを得ている．

閾値が 0.4 のときの ROUGE-1, ROUGE-2, ROUGE-L の再現率，適合率，F 値を表 4 に示す．なお，Gigaword のみから作成した訓練データ (Rush らの実験設定と同じ) で学習した結果（onlyGiga）も併せて掲載する．提案法により，上昇幅は小さいものの，データ量が減少した上で性能が向上したことが確認できる．

次に，図 3 より，ALL と同等の ROUGE スコアとなる

[*1] https://github.com/facebookarchive/NAMAS
[*2] Rush らのスクリプトに従い作成した評価データ，開発データから実験に用いるデータをそれぞれ無作為に選んだ．
[*3] https://github.com/mlpnlp/mlpnlp-nmt

表 3　各学習データにおける学習時間（h）．ALL は全訓練データでの，extr は提案法による学習データでの学習時間．

モデル	ALL	extr	時間短縮率
COPY	54	24	56%
ATTN	34	17	50%
SELE	60	35	42%

ようデータを削減すると，COPY は閾値 0.7 が，ATTN と SELE は閾値 0.6 がそれに該当することが分かる．このとき訓練データ量はそれぞれ 60%, 45%削減されている．これらのデータを用いてそれぞれのモデルを学習するのに必要な学習時間を表 3 に示す．学習時間は，プログラムを走らせ始めてから最終的に選ばれるモデルが出力されるまでの時間とした．結果をみると，全訓練データで学習する場合より大きく学習時間が減っていることが確認できる．特に COPY と ATTN では学習時間が 50%以上減っている．この結果は，提案法により学習に有効なデータが抽出できていることを示している．

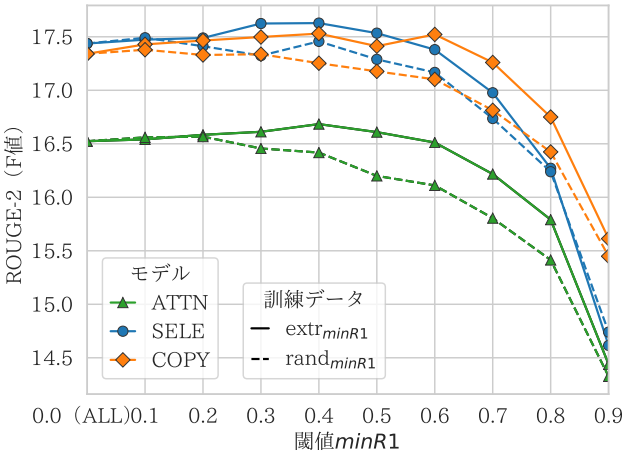

図 3　閾値を変化させた場合の ROUGE-2 スコアの変化．$extrD_{minR1}$ は提案法による学習データ．$randD_{minR1}$ は提案法と同じデータ数を訓練データから無作為に選んだ学習データ．左端 0.0(ALL) は全訓練データを用いた場合．

4.4　データ量を固定した比較実験

これまでは，閾値を変化させ，それに従いデータ量も変化する状況で実験を行った．しかし，より大量のデータが用意できる状況では厳しく閾値を設定しても十分なデータ量が確保できる．そこで，どの閾値においてもデータ量を 200 万文対に固定してデータ量の影響をなくしたうえで比較実験を行った．なお，閾値は 200 万文対以上を確保できる 0.6 以下を用いた．結果を図 4 に示す．

図より，閾値が高くなるにつれ，上げ幅は小さいものの性能が向上して行くことがわかる．特に，先ほどの実験で性能のピークを過ぎていた閾値 0.5 以降でも向上がみられる．同じ訓練データ数であっても閾値を厳しくする，つまり，抽出率を上げることで各手法の ROUGE スコアが向上していることから，抽出率が要約システムの性能に確実に影響を与えていると言える．

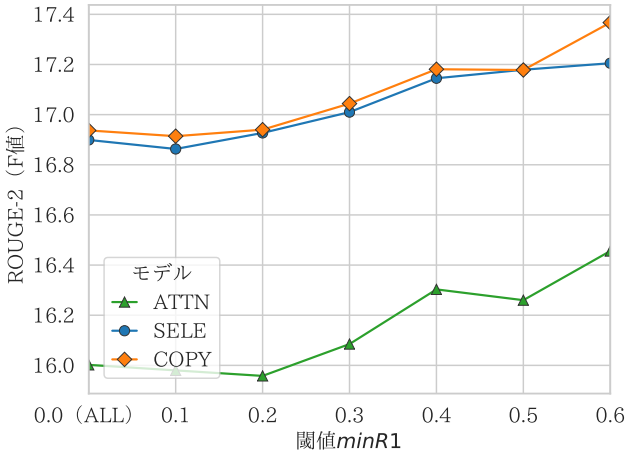

図 4　データ量を 200 万文対に固定して閾値を変化させた場合の ROUGE-2 スコアの変化．

5.　議論・分析

訓練データの抽出率を変化させると各要約手法の ROUGE が変化することは確認できた．次に，様々な抽出率で訓練した要約手法がどのような要約を生成するかを定量的に調べる．具体的には以下の 2 つの観点で調査を行う．

(1) システム要約の原文に対する抽出率，
(2) 正解（参照）要約における抽出率と ROUGE スコアの関係

本節では訓練データとモデルの関係を深く探るため，提案法により作成した抽出率の高い学習データだけでなく，逆に抽出率の低い（生成率の高い）学習データも作成し比較を行う．

5.1　生成率の高い学習データ

提案法の抽出率を用い，抽出率の低い，すなわち生成率の高い学習データを作成する．

閾値を minR1 とすると，得られる学習データ $abstD_{maxR1}$ は以下のように表せる．

$$extrD_{minR1}=\{(t,s)\in D|\mathrm{ROUGE}{-}1(t,s)\geq minR1\} \quad (3)$$

ここで，$\mathrm{ROUGE}{-}1(t,s)$ は参照文 t に対する原文 s の ROUGE-1（再現率），D は全データ対である．ただし，閾値 1 はデータ選択を行わないことを意味することに注意されたい．

生成率を高くする，つまり，maxR1 を小さくすると自動評価スコアが下がっていくと考えられる（実際のスコアは後述する表 6 を参照）．これは，要約モデルが原文の単語をなるべく再利用するようにしていることに対し，データは単語の再利用を許さないようになっていることが理由で

表 4　ROUGE による自動評価結果．太字は各モデルにおける最大値を示す．r，p，f はそれぞれ再現率，精度，F 値を示す．$extrD_{0.4}$ が提案法．

モデル	データセット	除去率	ROUGE-1			ROUGE-2			ROUGE-L		
			r	p	f	r	p	f	r	p	f
ATTN	onlyGiga	-	32.3	38.1	33.6	15.9	18.6	16.5	29.9	35.4	31.1
	ALL（$extrD_{0.0}$）	0%	32.3	38.2	33.6	15.9	18.7	16.5	30.0	35.5	31.2
	$extrD_{0.4}$	20%	**32.6**	**38.4**	**33.9**	**16.1**	**18.9**	**16.7**	**30.2**	**35.7**	**31.4**
SELE	onlyGiga	-	35.0	37.2	34.9	17.3	18.6	17.3	32.4	34.6	32.4
	ALL（$extrD_{0.0}$）	0%	35.3	37.3	35.1	17.5	18.7	17.4	32.8	34.7	32.6
	$extrD_{0.4}$	20%	**35.5**	**37.5**	**35.3**	**17.7**	**18.9**	**17.6**	**32.9**	**34.8**	**32.8**
COPY	onlyGiga	-	33.7	39.0	35.0	16.7	19.4	17.3	31.3	36.2	32.5
	ALL（$extrD_{0.0}$）	0%	33.8	**39.1**	35.1	16.7	19.4	17.3	31.4	36.2	32.5
	$extrD_{0.4}$	20%	**33.9**	**39.1**	**35.2**	**16.9**	**19.6**	**17.5**	**31.5**	**36.3**	**32.7**

ある．

5.2　訓練データの抽出性はモデル出力の抽出性にどう影響するか？

システム要約の抽出率の変化を分析する．ここからは 3 節で言及したコピー，ステムコピー，そしてその他（以降，生成）の 3 つの出力タイプを別々に扱い分析を行なっていく．ステミングには ROUGE で用いられているポーターステマー [24] を使用した．

まず図 5，図 6 に各手法における出力タイプの割合をマクロ平均で示す．比較のため，訓練データ全体における割合も合わせて提示する．図 5, 図 6 をみると，抽出率を高めた場合でも，生成率を高めた場合でも，学習データの変化に合わせて各手法の出力タイプも変化していることがわかる．抽出率を高めた場合では，学習データにおけるコピーとステムコピーが増え，生成が減ることに合わせ，各手法ともシステム要約におけるコピーが増え，ステムコピーが少し増え，生成が減っている．

生成率を高めた場合では，学習データにおけるコピーとステムコピーが減り，生成が増えることに合わせ，システム要約におけるコピーとステムコピーが減り，生成が増えている．1 節でも述べたとおり，学習データにおけるコピーの割合よりシステム要約におけるコピーの割合の方が基本的に高くなっている．また興味深いことに，生成率を高めた学習データには抽出率の高いデータ対がないにも関わらず，システム要約の抽出率はかなり高くなっている．これは現状の生成型要約手法は抽出率の高い要約を生成するよう設計されており，学習データの抽出性に関わらずシステム要約の抽出性が高くなりやすいことを示している．

5.3　正解要約における抽出率と ROUGE スコアの関係

続いてモデルが得意とする要約の種類，そしてその変化を分析する．要約文は様々な抽出率で作成することが可能であり，正解要約も様々な抽出率で作成されている．そこで抽出率 0.1 ごとに評価データを分割し，抽出率の異なる

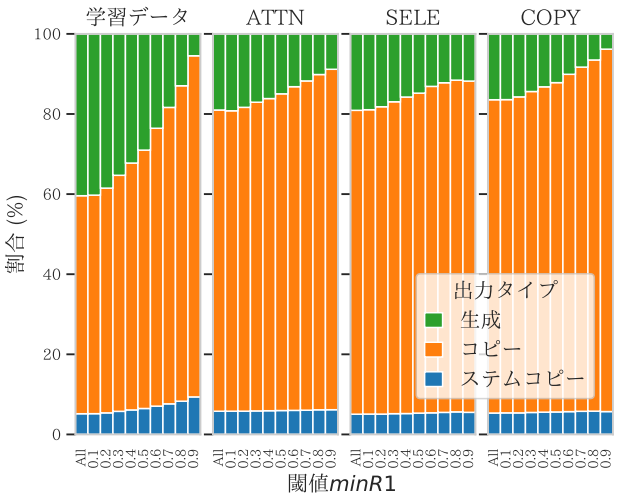

図 5　閾値を変化させ学習データ $extrD_{minR1}$ の抽出率を変えた場合の出力タイプの変化

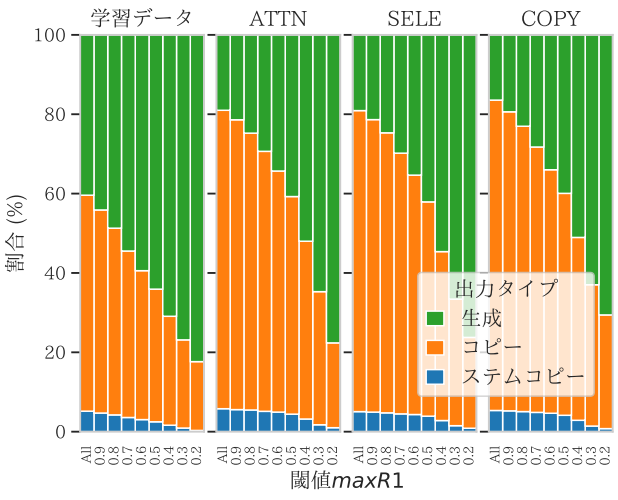

図 6　閾値を変化させ学習データ $abstD_{maxR1}$ の抽出率を変えた場合の出力タイプの変化

評価データごとに評価スコアを分析する．各評価データには，評価用コーパスの該当する抽出率をもつデータ対から 2000 文対を無作為に選んだ．結果を表 5，表 6 に示す．今回は抽出率の高い学習データには閾値 0.3, 0.5, 0.7

を，生成率の高い学習データには閾値 0.9,0.7,0.5 を使用し，ROUGE-2（F 値）を提示する．

表より，学習データの抽出率をあげると抽出率の高い評価データにおいて性能向上が顕著であり，学習データの生成率をあげると生成率の高い評価データにおいて多少ではあるが性能向上がみられる．これらの結果は学習データの抽出性を変化させることで，モデルが得意とする要約が変化したことを示している．ただし，生成率の高い評価データは根本的に極めて低いスコアとなっており，解くべき問題設定であるかには疑問が残る．

6. おわりに

抽出率の高い要約が適していると考えられる現状の文要約器のため，抽出率の低いデータ対を訓練データから取り除くデータ選択手法を提案した．自動評価による評価実験の結果，この提案法により文要約器の性能向上，そして学習時間の短縮ができることがわかった．また，訓練データの抽出率を変化させるとそれに合わせて文要約器の出力文の抽出率も変化すること，そして，訓練データの抽出率をあげると抽出率の高い要約が，生成率をあげると生成率の高い要約をうまく生成できることもわかった．今後の課題としては，人手評価を行うことがあげられる．

謝辞 本研究に対しコメントをくださった NTT コミュニケーション科学基礎研究所の平尾努氏に感謝申し上げます．

参考文献

[1] Rush, A. M., Chopra, S. and Weston, J.: A Neural Attention Model for Abstractive Sentence Summarization, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, Association for Computational Linguistics, pp. 379–389 (2015).

[2] Cao, Z., Wei, F., Li, W. and Li, S.: Faithful to the Original: Fact Aware Neural Abstractive Summarization, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, , the 30th innovative Applications of Artificial Intelligence , and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence, New Orleans, Louisiana, USA, February 2-7, 2018*, pp. 4784–4791 (2018).

[3] Gu, J., Lu, Z., Li, H. and Li, V. O.: Incorporating Copying Mechanism in Sequence-to-Sequence Learning, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Berlin, Germany, Association for Computational Linguistics, pp. 1631–1640 (2016).

[4] Junczys-Dowmunt, M.: Dual Conditional Cross-Entropy Filtering of Noisy Parallel Corpora, *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, Belgium, Brussels, Association for Computational Linguistics, pp. 888–895 (2018).

[5] Carpuat, M., Vyas, Y. and Niu, X.: Detecting Cross-Lingual Semantic Divergence for Neural Machine Translation, *Proceedings of the First Workshop on Neural Machine Translation*, Vancouver, Association for Computational Linguistics, pp. 69–79 (2017).

[6] Moore, R. C. and Lewis, W.: Intelligent Selection of Language Model Training Data, *Proceedings of the ACL 2010 Conference Short Papers*, Uppsala, Sweden, Association for Computational Linguistics, pp. 220–224 (2010).

[7] Khayrallah, H. and Koehn, P.: On the Impact of Various Types of Noise on Neural Machine Translation, *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, Melbourne, Australia, Association for Computational Linguistics, pp. 74–83 (2018).

[8] Belinkov, Y. and Bisk, Y.: Synthetic and Natural Noise Both Break Neural Machine Translation, *International Conference on Learning Representations* (2018).

[9] Koehn, P., Khayrallah, H., Heafield, K. and Forcada, M. L.: Findings of the WMT 2018 Shared Task on Parallel Corpus Filtering, *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, Belgium, Brussels, Association for Computational Linguistics, pp. 726–739 (2018).

[10] Xu, X., Dušek, O., Konstas, I. and Rieser, V.: Better Conversations by Modeling, Filtering, and Optimizing for Coherence and Diversity, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, Association for Computational Linguistics, pp. 3981–3991 (2018).

[11] Csáky, R., Purgai, P. and Recski, G.: Improving Neural Conversational Models with Entropy-Based Data Filtering, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp. 5650–5669 (2019).

[12] 松丸和樹， 高瀬翔，岡崎直観：含意関係に基づく見出し生成タスクの見直し，情報処理学会研究報告，Vol. 2019-NL-240, No. 1, pp. 1–8 (2010).

[13] See, A., Liu, P. J. and Manning, C. D.: Get To The Point: Summarization with Pointer-Generator Networks, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada, Association for Computational Linguistics, pp. 1073–1083 (2017).

[14] Zhang, F., Yao, J.-g. and Yan, R.: On the Abstractiveness of Neural Document Summarization, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, Association for Computational Linguistics, pp. 785–790 (2018).

[15] Kryściński, W., Paulus, R., Xiong, C. and Socher, R.: Improving Abstraction in Text Summarization, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, Association for Computational Linguistics, pp. 1808–1817 (2018).

[16] Grusky, M., Naaman, M. and Artzi, Y.: Newsroom: A Dataset of 1.3 Million Summaries with Diverse Extractive Strategies, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, New Orleans, Louisiana, Association for Computational Linguistics, pp. 708–719 (2018).

[17] Napoles, C., Gormley, M. and Van Durme, B.: Annotated Gigaword, *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, AKBC-WEKEX '12, Stroudsburg, PA, USA, Association for Computational Linguistics, pp. 95–100 (2012).

モデル	学習データ	抽出率で分割した評価データ											
		全データ	0.0〜	0.1〜	0.2〜	0.3〜	0.4〜	0.5〜	0.6〜	0.7〜	0.8〜	0.9〜	1.0
ATTN	ALL（$extrD_{0.0}$）	16.5	**0.1**	**1.0**	**3.0**	**6.5**	10.8	14.2	20.6	26.0	33.4	36.5	44.2
	$extrD_{0.3}$	**16.6**	**0.1**	0.8	2.8	6.4	**10.9**	**14.3**	**20.7**	26.2	33.8	36.5	44.4
	$extrD_{0.5}$	**16.6**	**0.1**	0.6	2.6	6.3	10.5	14.2	**20.7**	**26.5**	34.1	37.0	44.7
	$extrD_{0.7}$	16.2	0.0	0.5	2.3	5.8	9.8	12.8	19.1	26.1	**34.2**	**37.9**	**46.1**
SELE	ALL（$extrD_{0.0}$）	17.4	**0.1**	**1.1**	**3.1**	6.7	11.2	14.8	21.4	27.5	35.7	39.6	48.5
	$extrD_{0.3}$	**17.6**	**0.1**	0.9	2.9	**6.8**	**11.4**	**14.9**	**21.7**	27.7	**36.3**	39.9	49.1
	$extrD_{0.5}$	17.5	**0.1**	0.7	2.8	6.6	11.0	14.7	21.5	**28.0**	**36.3**	40.2	49.8
	$extrD_{0.7}$	17.0	0.0	0.5	2.4	6.0	10.1	13.3	19.9	27.3	36.1	**40.4**	**50.0**
COPY	ALL（$extrD_{0.0}$）	17.3	**0.1**	**1.0**	**3.3**	**7.0**	11.6	15.2	**21.9**	27.9	36.1	38.5	47.8
	$extrD_{0.3}$	**17.5**	**0.1**	0.8	3.0	6.9	**11.7**	**15.5**	21.8	**28.1**	36.3	38.5	48.2
	$extrD_{0.5}$	17.4	**0.1**	0.7	2.9	6.8	11.2	15.3	21.8	28.0	36.5	38.6	48.8
	$extrD_{0.7}$	17.3	0.0	0.5	2.5	6.3	10.3	14.0	20.6	27.7	**36.9**	**39.7**	**50.1**

表 5 抽出率ごとに分割した評価データにおける ROUGE-2（F 値）．学習データは ALL と抽出率を高めた学習データ $extrD_{minR1}$．

モデル	学習データ	抽出率で分割した評価データ											
		全データ	0.0〜	0.1〜	0.2〜	0.3〜	0.4〜	0.5〜	0.6〜	0.7〜	0.8〜	0.9〜	1.0
ATTN	ALL（$abstD_{1.0}$）	**16.5**	0.1	1.0	3.0	6.5	10.8	**14.2**	**20.6**	**26.0**	**33.4**	**36.5**	**44.2**
	$abstD_{0.9}$	16.2	0.1	1.1	**3.2**	**6.6**	**10.9**	14.1	20.5	25.9	32.7	35.0	40.5
	$abstD_{0.7}$	14.1	**0.2**	1.3	**3.2**	**6.6**	10.7	13.8	18.9	22.6	27.6	28.3	33.1
	$abstD_{0.5}$	10.7	**0.2**	**1.5**	**3.2**	5.7	9.2	11.5	14.2	16.3	19.2	17.9	22.5
SELE	ALL（$abstD_{1.0}$）	**17.4**	0.1	1.1	3.1	6.7	11.2	14.8	**21.4**	**27.5**	**35.7**	**39.6**	**48.5**
	$abstD_{0.9}$	17.1	0.1	1.1	3.2	**7.0**	**11.5**	**15.0**	**21.4**	27.2	35.1	38.1	44.3
	$abstD_{0.7}$	15.1	0.2	1.4	**3.4**	6.8	11.1	14.5	19.6	24.1	29.5	31.4	35.5
	$abstD_{0.5}$	11.2	**0.3**	**1.6**	**3.4**	5.9	9.5	12.0	14.6	17.2	20.1	19.1	23.4
COPY	ALL（$abstD_{1.0}$）	**17.3**	0.1	1.0	3.3	**7.0**	11.6	**15.2**	**21.9**	**27.9**	**36.1**	**38.5**	**47.8**
	$asbtD_{0.9}$	16.9	0.1	1.1	3.4	**7.0**	**11.7**	15.1	21.4	27.5	35.1	36.1	43.5
	$asbtD_{0.7}$	14.6	**0.2**	1.5	**3.5**	6.8	11.1	14.2	19.3	23.4	28.7	28.6	33.7
	$asbtD_{0.5}$	10.6	**0.2**	**1.6**	3.2	5.9	9.2	11.1	14.1	16.6	19.1	17.3	22.1

表 6 抽出率ごとに分割した評価データにおける ROUGE-2（F 値）．学習データは ALL と生成率を高めた学習データ $abstD_{maxR1}$．

[18] Consortium, L. D. and Company, N. Y. T.: *The New York Times Annotated Corpus*, LDC corpora, Linguistic Data Consortium (2008).
[19] Zhou, Q., Yang, N., Wei, F. and Zhou, M.: Selective Encoding for Abstractive Sentence Summarization, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada, Association for Computational Linguistics, pp. 1095–1104 (2017).
[20] Klein, G., Kim, Y., Deng, Y., Senellart, J. and Rush, A. M.: OpenNMT: Open-Source Toolkit for Neural Machine Translation, *ArXiv e-prints*.
[21] Tokui, S., Oono, K., Hido, S. and Clayton, J.: Chainer: a Next-Generation Open Source Framework for Deep Learning, *Proceedings of Workshop on Machine Learning Systems (LearningSys) in The Twenty-ninth Annual Conference on Neural Information Processing Systems* (2015).
[22] Hochreiter, S. and Schmidhuber, J.: Long Short-Term Memory, *Neural Comput.*, Vol. 9, No. 8, pp. 1735–1780 (1997).
[23] Luong, T., Pham, H. and Manning, C. D.: Effective Approaches to Attention-based Neural Machine Translation, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, Association for Computational Linguistics, pp. 1412–1421 (2015).
[24] Porter, M. F.: An algorithm for suffix stripping., *Program*, Vol. 14, No. 3, pp. 130–137 (1980).