Vol.2017-NL-232 No.3
2017/7/19

“長文質問”のための抽出型及び生成型要約

石垣 達也[1,a)]　高村 大也[1,b)]　奥村 学[1,c)]

概要：コミュニティ QA や学会等での質疑応答において用いられる質問は，核となる質問の他に補足的な情報も付加され，ときに複数の文で構成されることもある．このような “長文質問”は，質問の受け手にとって，要旨の把握が難しい．そこで，本研究では長文質問を端的に表現する 1 文に要約する課題を提案する．また，コミュニティ QA の質問本文，タイトルのペアを長文質問と要約の対とみなし，抽出型及び生成型の要約モデルを学習し，性能を評価する．

1. はじめに

“質問する”という行為は，コミュニティ QA への質問投稿，学会の質疑応答，メールによる問い合わせなど，多くの状況で用いられる．このような状況で発せられる質問には，核となる質問の他に，質問をするに至った背景等の補足的な情報も加えられ，ときに複数文で構成されることもある．質問が長くなるにつれ，質問の受け手にとっては質問の要旨を把握することが難しくなる．また，それにより質問者にとっても適切な回答を得ることが難しくなる．本研究では，このような “長文質問”を，内容を端的に表現する 1 つの質問に要約する課題を提案する．

コミュニティ QA の Yahoo! Answers[*1]から抜粋した長文質問の例を表 1 に示す．この例において，核となる質問は “塩素 (chlorine) によって頭髪の染料が落ちるか否か”である．しかし，長文質問においては質問者が水泳をするといった情報や，水泳を行う頻度など質問の背景となる補足的な情報が付与される．このような例では，核となる質問だけが提示される場合に比べ，質問の要旨の把握が難しくなる．長文質問を，内容を端的に表現する 1 つの質問に要約することができれば，回答者の質問理解の手助けになり，質問者は希望する回答を得やすくなる．また，長文質問を読む前に要旨を把握することができれば，長文質問を読む際にも焦点を絞ることができ，理解が深まる．

既存の要約課題で用いられる手法は抽出型，生成型に大きく分類できる．抽出型の手法では，入力文書に含まれる文のうち要約に適した文を選択し出力する．生成型の手法では，入力文書から文を選択するのではなく，入力文書に出現しない表現も用いて要約を生成する．抽出型，生成型どちらの既存研究においても入力として質問を扱う研究は少ない．田村ら [1] は複数文で構成される質問にも対応する質問応答システムを実現するためのサブタスクとして，長文質問から重要文を抽出する手法を提案した．この手法は，複数文の質問を受け付ける質問応答システムにおける質問タイプ同定の前処理を想定しており，表 1 の長文質問における “will it or not?”のように，質問タイプ同定に用いるには有用であるが要約としてそのまま提示すると情報が不足する文が抽出されることがある．機械翻訳や要約課題への応用が報告されている encoder-decoder を用いたモデルでは，文書とその要約の対から end-to-end にテキスト生成モデルを構築することができる．encoder-decoder の学習には長文質問とその要約の対が大量に必要であるが，そのようなデータセットは我々の知る限り存在しない．

表 1　“長文質問”と要約の例

長文質問:
im a swimmer for my school swim team and i have pratice 2 hours a day for 5 days a week . i would like to dye my hair black (it is dark brown now) but i am not sure if it will get stripped by the chlorine . will it or not ?
要約例:
will my hair get stripped by the chlorine ?

本研究では，コミュニティ QA である Yahoo! Answers に投稿される質問本文とそのタイトルを，長文質問とその要約の対とみなし，抽出型および生成型の要約モデルを構築する．Yahoo! Answers データセットには，長文質問とその要約とはみなすことのできない質問本文/タイトルの対も含み，そのまま要約モデルの学習に用いることができない．そのため，要約対とみなすことのできる事例および

[1] 東京工業大学
[a)] ishigaki@lr.pi.titech.ac.jp
[b)] takamura@pi.titech.ac.jp
[c)] oku@pi.titech.ac.jp
[*1] https://answers.yahoo.com

みなすことのできない事例の特徴について事例分析を行い，フィルタリング規則を構築する．また，フィルタリングして得た長文質問とその要約の対において，どのような方法を用いて要約が生成されているか分析を行う．最後に，抽出型および生成型の要約モデルを Yahoo! Answers のデータから作成した長文質問とその要約の対から学習し性能を評価する．

2. 関連研究

テキスト要約は自然言語処理の分野において古くから研究されている課題の 1 つである．要約課題の設定としては，要約の Shared Task である Document Understanding Conference (DUC)[*2]のように新聞記事や科学論文を入力として扱うことが多い．一方で，会話文書 [2] やメールスレッド [3][4] の要約課題など，新聞記事とは異なる特徴を持った文書を入力とする課題も提案されている．これらの研究とは異なり，本研究では入力として長文質問を想定する．

要約課題へのアプローチは，大きく “抽出型” もしくは “生成型” へ分類される．

抽出型の要約手法では，入力文書から文などの単位を抽出し，要約として提示する．多くの研究の問題設定では，入力文書と文長を入力として受け取り，抽出文を組み合わせ文長に合わせた要約を生成する．本研究では出力文は 1 文という制限を用いるが，文長の制限はない．抽出型要約手法としてよく知られる TextRank[5] では，文を頂点，文間の類似度をエッジの重みとしたグラフ構造を用いて文書を表現する．このグラフにおいて，より多くの頂点から重みの大きなリンクが張られている文は重要であるという仮定に基づき重要文を決定する．このアプローチでは，出力が必ずしも質問になるとは限らず，長文質問の要約という本研究の問題設定にそのまま適用すると適切な文を出力することができないことがある．

長文質問から 1 つの質問を抽出する問題設定としては，田村ら [1] の研究がある．田村らは複数文で構成される質問から，質問タイプ同定のために重要な文を抽出するサブタスクを扱った．例えば，表 1 の長文質問における最後の文 “will it or not?” では，質問タイプが Yes/No Question であると同定するには有用である．しかし，そのまま要約として提示しても，長文質問の要旨を把握することはできない．このように，質問タイプ同定の手がかりとしては有用な文であっても，要約としてそのまま提示するには不適切な文が存在する．

生成型の要約手法では，入力文書から文などの単位を抽出するのではなく，一度中間表現に変換したのち言語生成モデルを用いて要約を生成する．そのため，原文書には含まれない表現が出力されることもありえる．生成的な要約手法においては，テンプレートベースの手法 [2] に加え，近年では，機械翻訳向けに提案された encoder-decoder を文の要約課題に応用する手法 [6][7] が積極的に研究されている．encoder-decoder を用いた要約手法においては，テキストとその要約の対を翻訳対と考え，encoder-decoder を学習する．入力系列の特定箇所に着目しながら出力を生成するか考慮する注意機構を加えたモデル [6][8] や，入力に含まれる語を出力に含める CopyNet[9] などの亜種も積極的に研究されている．このアプローチを用いて要約モデルを学習するには，長文質問とその要約の対が大量に必要であるが，我々の知る限りそのような大規模データは存在しない．そこで，本研究ではコミュニティ QA である Yahoo! Answers に投稿された質問本文とタイトルの対を，長文質問とその要約の対とみなし，大規模なデータセットの作成を試みる．

表 2 質問本文の文数と要約とみなせる対の関係

質問本文の文数	要約とみなせる対の割合
1	3/20
2	8/20
3	14/20
4	15/20
5	15/20

3. データ分析

本研究では，まずコミュニティ QA の Yahoo! Answers の提供するデータセット “Yahoo! Answers Comprehensive Questions and Answers version 1.0”[*3]を対象にデータ分析を行う．このデータセットには，2005 年 6 月 28 日から 2007 年 10 月 25 日までに投稿された 4,483,032 の質問が格納されている．データには質問本文とそれに対応するタイトルが含まれ，どちらもユーザが自由に記述したものである．そのため，質問本文とタイトルの対には，長文質問とその要約とはみなすことのできる対およびできない対の両方が存在する．要約モデルを学習するためには，要約とはみなすことのできない事例をフィルタリングする必要がある．さらに，長文質問が抽出的に要約可能であるか，生成型による要約が必要であるかも明らかではない．そこで，まず本研究では以下の 2 点に着目し事例分析を行う．

(1) 要約対とみなすことのできない質問本文/タイトル対の特徴はなにか
(2) 要約対とみなすことのできる質問本文/タイトル対において，どのような方法で要約質問が生成されていると考えられるか

3.1 質問本文の長さ

要約対とみなすことのできない質問本文/タイトルの特徴を明らかにするために，まず質問本文の文数との関係に

[*2] http://duc.nist.gov

[*3] https://webscope.sandbox.yahoo.com/

表 3 抽出的および生成的に要約を作っている対の数

要約対とみなすことができない	5/20
抽出型の手法で要約ができる	8/20
生成型の手法が必要	7/20

着目した．質問本文が 1 文から 5 文の質問本文/タイトルの対をランダムに 20 対ずつ抽出し，要約対とみなせるか否かを人手で判定した．要約対とみなせるか否かの判定においては，以下の 2 つの基準をどちらも満たすものを要約対とした．質問本文と回答，タイトルと回答の両方が意味の通るペアであれば要約対であると判定した．

(1) 質問本文と回答が整合している
(2) 質問のタイトルと回答が整合している

表 2 に本文の文数と要約対とみなすことのできる対の数との関係を示す．

この分析から，質問本文の文数が 2 文以下である場合に，要約対とはみなすことのできない事例が増える．また，文数が 3 文以上では，要約とみなせる対の割合がほぼ一定になる．本文の文数は手がかりの 1 つとして用いることができると考えられる．

3.2 質問本文とタイトルの名詞の重複

要約対とみなすことのできないペアの例を以下に示す．

タイトル: Why is there often a mirror in an elevator?

質問本文: What is the history behind it?

この例において，質問本文はタイトルの質問に関連する別の質問になっており要約対とみなすことができない．質問本文の文数が 2 文以下である場合，このような事例が多く要約対とみなせない質問本文/タイトル対が増える．また，この例においては質問本文中に “mirror” “elevator” といった語が出現せず，質問本文のみを提示しても質問を理解することができない．このように，タイトル中に含まれる名詞が質問本文に出現しない場合においては，質問本文/タイトルを要約対とみなせない．以上の分析より，質問本文の文数に加え，タイトルと質問本文の名詞の重複が，要約対であるか否か判定するための手がかりになると考えられる．

3.3 抽出的 vs. 生成的

次に Yahoo! Answers の質問本文/タイトルを，長文質問とその要約の対とみなした場合に，どのように要約が生成されているか分析する．具体的には質問本文の長さが 3-5 文の対をランダムに抽出し，以下の 3 つに人手で分類した．

- 本文とタイトルは要約対とみなすことができない
- 抽出型の手法で要約ができる
- 生成型の手法が必要

分類した結果を表 3 に示す．また，長文質問とそのタイトルの代表的な例を表 4 に示す．

20 事例のうち 5 事例は，タイトル中の代名詞が本文中の語を照応したり，“please help!” など質問本文の内容を表現していない事例であり，要約対とみなすことができない．例えば，表 4 の質問本文例 1 は “シューズを汚してしまい困っている” といった内容が記述されているが，タイトルは “please help!” となっており，質問本文の内容をタイトルが表現していない．

20 対中 8 対は，抽出的に要約を作ることのできる事例であった．この 8 対のうち 6 対は，抽出的に要約を作ることができるが，実際のタイトルではさらに単語除去や言い換えを行っている．このような事例として表 4 の質問本文例 2 とそのタイトルを例示する．この例では，末尾の質問文を抽出することで要約を出力できる．しかし，タイトルでは “get rif of” が “remove” に言い換えられたり，“winflexer” という語が除去されたりと，さらに短くなっている．このように抽出的に要約を作ることのできる質問本文でも，実際のタイトルでは生成的に要約が作られている質問本文/タイトルの対が存在する．

20 対中 7 対については，抽出による手法では要約を作ることができず，生成的な手法を用いる必要がある．

生成的に要約を作っている対はさらに，以下の 2 つに分類できる．

- 核となる質問文から前に出現した語を照応する
- 長い補足説明から短い質問が続く

質問本文例 3 の質問本文では，核となる質問 (“do you like it ?”) から前に出現した語 (“the simpsons”) を照応している．そのため，そのまま抽出して要約として提示すると照応の問題が発生する．タイトルでは照応を適切に解消している．質問本文例 4 では，クッキーの調理について説明を加えた後に，“Why?” という短い質問文に接続されている．このような例では，末尾の文を抽出するだけでは要約として適切な文が抽出できない．タイトルでは複数文にまたがる情報をつなぎ合わせ，文書全体を要約している．

特に表 4 の質問本文例 3 および 4 については，抽出型の手法で要約を生成するのが難しく，生成型の手法を用いる必要がある．

4. データセットと要約手法

抽出的な手法，生成的な手法を用いて要約を生成し，出力を比較した．本節では，実験に用いるデータセットの作成方法および比較手法について説明する．

4.1 データセットの作成

本研究ではコミュニティ QA に投稿される質問本文とそのタイトルを，長文質問とその要約の対とみなし，要約モデルを学習する．今回はコミュニティ QA のデータとして，Yahoo! Answers の提供するデータセット “Yahoo!

表 4 Yahoo! Answers dataset 内の代表的な質問本文/タイトルの対

質問本文例 1(タイトルが質問本文の内容を表現しない例):
i accidentally spilled popcorn butter on my leather hospital shoe.
it has dark spots now and i don't know how i could get it off.
...
タイトル: please help!
質問本文例 2(抽出することで要約として適切な文を抽出できる．正解ではさらに単語除去や言い換えを行う):
i have annoying winfixer pos ups . have tried all sorts of ad removal programs but no success .
how can i get rid of annoying winfixer pop ups ?
タイトル:
how to remove annoying pop ups ?
質問本文例 3(複数文にまたがる情報を 1 つの質問にまとめる):
the simpsons is one of the funniest shows ever . its one of my favorites . do you like it ?
タイトル:
do you like the simpsons ?
質問本文例 4(長い補足説明から短い質問が続く例):
I've experimented with various recipes as well as oven temperature. my cookies always turn out thin.
I want the substantial, gooey kind–crips outside, chewy inside. Why?
タイトル:
Why do my chocolate chip cookies always turn out flat?

表 5 質問本文に含まれる質問文の数とデータセット内での割合

質問文の数	0 文	1 文	2 文	3 文	4 文	5 文
割合 (%)	32.3	40.1	18.0	7.1	2.0	0.6

Answers Comprehensive Questions and Answers version 1.0"を用いる．このデータセットには，2007 年 10 月 25 日までに投稿された質問が 4,483,032 格納されている．データ分析から，データセットには質問本文とタイトルが要約とみなすことのできない事例が含まれていることがわかった．

そこで，以下のフィルタリング規則を用いて，長文質問とその要約とみなすことのできないペアを取り除く．

複数文質問タイトル タイトルを文分割し，タイトルが複数文から構成される対をフィルタリング

長いタイトル タイトルが 16 語よりも多い対をフィルタリング

短いタイトル タイトルの語が 3 語以下の対をフィルタリング

名詞の重複 タイトルの一般名詞が本文に出現しない対をフィルタリング

本文の文数 本文の文数が 2 文以下もしくは 6 文以上の対をフィルタリング

なお，データセット内には英語以外のデータも含まれるが，今回は英語とタグ付けられたデータのみを対象にする．フィルタリング後の 251,420 対を要約モデルの構築および性能評価に用いる．質問本文に含まれる質問文の数と，データセット内での割合を表 5 に示す．

4.2 抽出的な手法

抽出的な手法として，規則による手法，機械学習による手法で要約を生成した．

4.2.1 規則を用いる手法

規則による手法として，“先頭文” “先頭質問文” “末尾質問文”を出力する 3 つの手法を用いた．質問文の判定においては末尾語が “?”であれば質問文と判定している．

4.2.2 機械学習を用いる手法

機械学習を用いる手法には，回帰モデルを用いる手法，分類モデルを用いる手法の 2 つを用いた．

回帰モデルを用いた手法では，入力の各文に対し，回帰モデルを用いてその文を抽出した場合の ROUGE の予測値を計算する．その後，ROUGE の予測値が最大になる文を出力する．回帰モデルの学習に用いる正解の ROUGE 値には，データセット内のタイトルを正解要約とみなした場合の ROUGE-2 の F 値を用いた．回帰モデルにはサポートベクトル回帰 (SVR) を用いた．

分類モデルを用いた手法においては，回帰モデルと同様にタイトルを正解要約とみなし，各文の ROUGE-2 の F 値をあらかじめ計算する．文書内で 4 語以上の質問文のうち ROUGE 値が最大になる文を正例，それ以外の文を負例として二値の分類モデルを学習した．そのため，負例には 3 語以下の質問文や平叙文を含む．4 語以上の質問文であっても，他に ROUGE 値がより高い文があれば負例とラベル付けされる．分類モデルにはサポートベクトルマシン (SVM) を用いた．SVM が入力の複数の文を正例と判定した場合には，それらの文のうち先頭文を出力する．SVM が文書内のすべての文を負例と判定した場合には，先頭の質問文を出力する．質問文が文書内に存在しない場合には質問文を出力する．質問文の判定は，規則を用いる方法と同様に末尾語が “?”であるかという規則を用いる．

回帰モデルおよび分類モデルの学習には以下の素性を用いた．

- 単語
- 文長
- 先頭文
- 先頭質問文
- 他の質問文の存在

素性はすべて 0 もしくは 1 の二値で表現する．単語素性には訓練データ中の単語のうち出現回数が 5 回以上のものを用いる．文長素性には，文長が 4 語以下，10 語以上，15 語以上の素性を用いる．なお，分類モデルの学習には線形カーネルを用い，パラメータ C には開発セットでの性能がもっとも良くなる値を選択した．

4.3 生成的手法

本研究では生成的な手法として encoder-decoder，注意機構を追加した encoder-decoder，CopyNet の追加された encoder-decoder を学習し，比較を行う．

長文質問は 3-5 文から構成され，機械翻訳などの問題設定に比べ入力系列が長い．そのため，注意機構を追加したモデルを用い，入力系列のうちデコード時の手がかりになる箇所に重みを付けながら質問を生成する．質問要約においては，入力質問に出現する語が出力にも含まれることが多い．そこで，出力に入力質問の語を用いる CopyNet を用いたモデルとの比較も行う．

encoder-decoder および注意機構付きのモデルには，Luong らの手法 [8] を用いる．CopyNet の付いたモデルについては Cao[9] らの手法を用いる．以下にこれらのモデルを簡単に説明する．

4.3.1 encoder-decoder

encoder-decoder モデルは，encoder および decoder という 2 つの要素から構成される．encoder は，入力の長文質問 $\boldsymbol{x} = x_1, ..., x_n$ を 1 単語ずつ受け取り，連続値ベクトルによる内部状態 $\boldsymbol{h}_\tau$ に逐次変換する:

$$\boldsymbol{h}_\tau = f(x_\tau, \boldsymbol{h}_{\tau-1}). \tag{1}$$

f は Recurrent Neural Network (RNN) で用いる任意の活性化関数で，本研究では，encoder-decoder および注意機構付きの encoder-decoder では Long Short Term Memory (LSTM)[10] を用い，CopyNet モデルには Gated Recurrent Unit (GRU)[11] を用いる．

decoder は 1 つ前のタイムステップ時点で生成された単語および内部状態を受け取り，現在のタイムステップ時点での内部状態 $\boldsymbol{s}_t$ を計算する．計算した内部状態を用いて softmax 関数により語 y_t の生起確率を計算できる．なお，Decode の初期状態には入力質問中の単語をすべて Encode したときの最終状態 $\boldsymbol{h}_n$ を用いることにする:

$$\boldsymbol{s}_t = f(y_{t-1}, \boldsymbol{s}_{t-1}), \tag{2}$$

$$p(y_t|y_{<t}, \boldsymbol{x}) = softmax(g(\boldsymbol{s}_t)). \tag{3}$$

$\boldsymbol{x}$ が与えられた上での出力系列 $\boldsymbol{y} = y_1, ..., y_m$ が生起する条件付き確率は，以下のように出力単語の生起確率の積に分解される:

$$p(\boldsymbol{y}|\boldsymbol{x}) = \prod_{t=1}^{m} p(y_t|y_{<t}, \boldsymbol{x}). \tag{4}$$

学習時には訓練データにおける対数尤度を最大化する:

$$\log p(\boldsymbol{y}|\boldsymbol{x}) = \log \sum_{j=1}^{m} p(y_t|y_{<t}, \boldsymbol{x}). \tag{5}$$

学習後は，入力の長文質問 $\boldsymbol{x}$ に対しビーム探索により出力 $\boldsymbol{y}$ を求める．

4.3.2 注意機構付き encoder-decoder

前節で述べた encoder-decoder は，デコード時に直前の内部状態 $\boldsymbol{s}_{t-1}$ と直前の出力単語 y_{t-1} の情報のみを用いて逐次デコードしていく．注意機構付きの encoder-decoder では，デコード時に $\boldsymbol{s}_{t-1}$ や y_{t-1} だけでなく，エンコーダ側のどの単語に着目するかを考慮した重み付き文脈ベクトル $\boldsymbol{c}_t$ を考える．

$$\boldsymbol{c}_t = \sum_{\tau=1}^{n} \alpha_{t\tau} \boldsymbol{h}_\tau. \tag{6}$$

$\alpha_{t\tau}$ は，t 番目の単語を出力する際に入力質問の τ 番目の単語に与えられる重みで，以下のように softmax 関数を用いて計算される:

$$\alpha_{t\tau} = \frac{exp(\boldsymbol{s}_t \cdot \boldsymbol{h}_\tau)}{\sum_{h'} exp(\boldsymbol{s}_t \cdot \boldsymbol{h}')}. \tag{7}$$

入力側の重み付き文脈ベクトル $\boldsymbol{c_t}$ と $\boldsymbol{h}_t$ を用いて，内部状態 $\tilde{h}$ を以下のように計算し，softmax 関数で確率値を出力する:

$$\tilde{\boldsymbol{h}} = tanh(\boldsymbol{W}_c[\boldsymbol{c}_t; \boldsymbol{h}_t]). \tag{8}$$

$$p(y_t|y_{<t}, \boldsymbol{x}) = softmax(\boldsymbol{W}_s \tilde{\boldsymbol{h}}_t). \tag{9}$$

4.3.3 CopyNet を用いた encoder-decoder

要約課題においては，入力に含まれる語が出力にも出現することが多い．そこで，CopyNet を備えた decoder を用いて入力質問に含まれる語を用いた要約の生成を試みる．Cao ら [9] の提案した decoder は，単語の生起確率の計算を以下のように行う:

$$p(y_t|y_{<t}, \boldsymbol{x}) = \begin{cases} \alpha_{t\tau} & (if\ y_t = x_\tau) \\ 0 & (otherwise). \end{cases}$$

この式において，入力文書に含まれる語の生起確率には重み付き文脈ベクトル $\boldsymbol{c}_t$ の重みが用いられる．一方で，入力文書に出現しない語の生起確率は 0 になる．これにより，要約は入力の単語を用いて要約が生成される．

表 6　ROUGE-2 による評価

	再現率	F 値
先頭文	39.4	27.0
末尾質問文	42.6	33.9
先頭質問文	45.3	34.5
分類モデル	44.3	35.1
回帰モデル	44.7	29.7
EncDec	3.5	2.6
EncDec+Attn	38.5	38.5
CopyNet	45.9	39.2

5. 評価と考察

本研究では ROUGE[12] を用いた自動評価，人手評価及び出力の定性的な分析を行う．

5.1 実験設定とモデルの学習

データセット (251,420 事例) は訓練セット，開発セット，評価セットに 9:0.5:0.5 に分割した．SVM および SVR の学習におけるパラメータ C の値は，開発セットにおける性能がもっとも良くなる値を採用した．

encoder-decoder の学習における単語埋め込み層，隠れ層の次元はそれぞれ 256，バッチサイズは 64 に設定した．また，出現回数が 1 回以下の語は特別なトークン UNK で置換した．文末には文末を表すトークン EOS を付与している．開発セットでの損失関数が最小になった 9 イテレーション目の結果で評価する．また，encoder-decoder がデコードに失敗し末尾の EOS トークンを 20 単語以内に出力しなかった場合は，先頭質問文を出力している．

5.2 ROUGE による評価

今回の問題設定では，出力は 1 文であるという制約はあるが，文長の制約はない．しかし，出力としてはより短く内容を端的に表現した質問であることが望ましい．そのため，ROUGE 値による自動評価においては，ROUGE-2 の再現率による評価だけでなく，F 値を用いた評価を行う．

各比較手法での ROUGE-2 の再現率および F 値による値を表 6 に示す．比較には規則を用いた抽出的な手法として，先頭文，末尾質問文，先頭質問文，機械学習を用いた手法として SVR による回帰モデル，SVM による分類モデルの結果を示す．生成的な手法として encoder-decoder および，注意機構を付加したモデル，CopyNet を用いた手法を示す．

規則を用いた抽出的な手法においては，既存要約課題において有用なベースラインとして知られる先頭文を出力する手法は再現率/F 値ともにもっとも低い値となった．長文質問の要約においては，質問文を出力することで先頭文ベースラインを上回る ROUGE を得ることができる．複数の質問文が存在する場合には，先頭質問文を出力することで，再現率/F 値ともに末尾の質問文を出力するよりも高い値を得る．

機械学習を用いた手法は分類モデルの方が回帰モデルよりも良い性能を示した．分類モデルの性能は先頭質問文を出力する場合と大きな差が見られなかった．今回用いたモデルにおいては，分類器が入力の文をすべて負例 (要約として用いることが適切ではない文) と判定した場合は，先頭質問文を出力する．入力文書に含まれる質問は 1-2 文程度であり，分類器を用いたモデルと先頭質問文を出力するモデルはほぼ同じ出力をしている．

生成的な手法では，注意機構や CopyNet を追加した encoder-decoder モデルは抽出的な手法よりも F 値において良い性能を示した．注意機構を持たない encoder-decoder においては，入力の系列の特徴を捉えた中間ベクトルが生成できず，decoder がほぼすべての入力に対し同じ質問を出力したため，極端に低い ROUGE を示した．注意機構を追加することにより，入力系列のエンコードが適切に行われない問題は改善された．CopyNet を追加することにより，再現率/F 値はさらに高い値を示した．

抽出的な手法は全体として長い系列を出力する傾向にある．そのため，再現率による評価では生成的な手法よりも高い値を示すが，F 値での評価においては低い値を示す．

5.3 人手評価

人手評価はクラウドソーシングサービスである Crowdflower[*4]を用いて行った．評価者には，質問本文と以下の 4 つの手法で出力した要約を提示し，very poor (1 点)，poor (2 点)，barely acceptable (3 点)，good (4 点)，very good (5 点) の 5 段階で，各出力に点数を付与した．

- 人間によるタイトル
- 先頭質問文
- 分類モデル
- CopyNet

評価指標には要約課題の Shared Task である DUC における評価基準を基に，“フォーカス”および “文法性”の 2 つを用いた．フォーカスによる評価は，質問本文の内容が要約でも適切に表現されていれば，より高いスコアを付けるよう作業者に依頼した．文法性についても同様に，文法的に正しい出力により高いスコアを付けるよう依頼した．評価者による評点の後，それぞれの手法がより良いと判定された回数をカウントした．評価者には明らかに文法性およびフォーカスに間違いを含む要約を提示し，不正解の作業者の回答は排除した．評価データには自動評価で用いたデータからランダムに 100 事例を抽出し，1 事例に対し 3 人の作業者が評価を行った．

Yahoo! Answers のタイトルの平均文長は 8.8 単語であ

[*4] https://www.crowdflower.com

表 7 人手評価-フォーカス-(先頭質問文が 8 語以下)

	人間	先頭質問文	分類モデル	CopyNet
人間	-	78	78	68
先頭質問文	21	-	3	14
分類モデル	21	1	-	15
CopyNet	25	24	27	-

表 8 人手評価-文法性-(先頭質問文が 8 語以下)

	人間	先頭質問文	分類モデル	CopyNet
人間	-	43	43	42
先頭質問文	17	-	2	17
分類モデル	17	1	-	16
CopyNet	24	18	19	-

表 9 人手評価-フォーカス-(先頭質問文が 9 語以上)

	人間	先頭質問文	分類モデル	CopyNet
人間	-	42	45	53
先頭質問文	43	-	5	40
分類モデル	42	3	-	40
CopyNet	33	12	14	-

表 10 人手評価-文法性-(先頭質問文が 9 語以上)

	人間	先頭質問文	分類モデル	CopyNet
人間	-	33	31	48
先頭質問文	38	-	5	31
分類モデル	40	7	-	35
CopyNet	34	11	12	-

る．本研究では文長の制限はないが，正解要約に近い文長で出力することが望ましい．encoder-decoder に基づくモデルは訓練データの文長の長さで出力しやすいことが知られ，出力の平均文長は 8.3 単語であった．一方で，先頭質問を出力した場合の平均文長は 10.8 単語であった．そこで，人手評価では先頭質問が 8 単語以下の事例と 9 単語以上の事例を分けて評価を行い，各手法の違いを分析する．

人手評価の結果を表 7 から表 10 に示す．この表では，2 つの手法を比較しもっとも左カラムに示された手法が良いと判定された回数をカウントしている．

先頭質問文が 8 単語以下の事例での評価においては，先頭質問文を抽出するよりも CopyNet を組み合わせた手法の方が良いと判定された回数が多い．8 単語以下の事例においては，encoder-decoder の方が先頭質問文より，フォーカスの正しい情報を埋め込むことができていると考えられる．先頭質問文と分類モデルでは，ほとんど差がないことから，分類モデルの出力は先頭質問文とほぼ同様になっていることが，人手評価の結果からも分かる．CopyNet と人間を比較すると，平均文長がほとんど変わらないにもかかわらず人間の方が良いと評価されている．人間によるタイトルにおいては，より短い文に質問本文のフォーカスに近い内容が埋め込まれていることが分かる．

先頭質問文が 8 単語以下の事例での文法性の評価でも，人間は他の手法よりも高い評価を得た．それ以外の 3 手法は文法性の面でほとんど差が見られなかった．

表 11 各手法の出力例

質問本文
the simpsons is one of the funniest shows ever . its one of my favorites . do you like it ?
人間 : Do you like the simpsons?
先頭質問文 : do you like it ?
分類モデル : do you like it ?
EncDec+Attn: do you like UNK ?
CopyNet : do you like the simpsons ?

先頭質問が 9 単語以上の事例においては，抽出的な手法のほうがより多くの情報を詰め込むことができ，フォーカスによる評価が高くなる．それでもなお，人間の要約が高く評価されている．したがって，人間は先頭質問文よりも短い要約の中に先頭質問文以上にフォーカスの正しい内容を埋め込んでいるといえる．9 単語以上の先頭質問は encoder-decoder の生成する要約よりも，フォーカスの正しい内容を出力することができる．

表 10 に示す，文長が 9 単語以上の事例での文法性に関する評価では，先頭質問文の文法性の方が人間の付与したタイトルよりも高いと判定された．人間のタイトルでは文法性を犠牲にして，より短い要約を生成していると考えられる．先頭質問文が 9 単語以上と長い事例については，文法的に正しく，より多くの情報を 1 つの文に埋め込んでいる．

5.4 出力例と定性的な分析

表 11 に出力の例を示す．抽出的な手法である先頭質問文や分類モデルの出力では，末尾の核となる質問に含まれる代名詞が照応する先が明らかでない．そのため，これらの出力のみから長文質問の内容を把握することができない．このような事例は，長文質問の前半部分で補足的な説明が行われ，その後核となる質問に続く例に多く見られる．抽出的なモデルでは照応など，複数文にまたがる情報を要約に埋め込む必要がある場合には困難が生じることが分かる．一方，CopyNet の出力では，it の部分が適切に “the simpsons” という照応先に置換され，照応の問題が解決されている．しかし，注意機構を追加したモデルでは，出力に未知語を表現するトークン (UNK) が含まれ，そのまま提示できる要約を出力できていない．encoder-decoder の既存研究 [13] では，出現頻度の低い内容語が出力されるべき部分が UNK トークンで出力される現象が報告されており，本研究でも同様の結果となった．この問題は CopyNet により大きく改善された．

6. まとめ

本研究では長文質問を入力とし，質問内容を端的に表現する 1 つの質問を出力する要約課題を提案した．Yahoo!

Answers dataset に含まれる質問本文とタイトルの対を，長文質問とその要約の対とみなし分析を行った．分析より，長文質問の要約においては，抽出的に要約を生成することのできる例，生成的に要約を生成する必要のある例がどちらも存在することがわかった．また，Yahoo! Answers の質問本文とタイトルをフィルタリングし，長文質問とその要約を格納したデータセットを作成した．

このデータセットを用い，抽出型/生成型の要約モデルをそれぞれ構築し評価を行った．その結果，抽出型の手法はより長い出力を許容する問題設定においては性能が高く，生成型は 8 単語以下などの文長制限がある設定において抽出型よりも良い性能を示した．出力の定性的な分析では，生成型の要約器が照応の問題を正しく解決している例など，抽出型の欠点を改善する出力を確認した．

謝辞

本研究は JST さきがけ JPMJPR1655 の支援を受けたものです．

参考文献

[1] Tamura, A., Takamura, H. and Okumura, M.: Classification of Multiple-Sentence Questions, *Proceedings of IJCNLP-05*, pp. 426–437 (2005).
[2] Oya, T., Mehdad, Y., Carenini, G. and Ng, R.: A Template-based Abstractive Meeting Summarization: Leveraging Summary and Source Text Relationships, *Proceedings of INLG2014*, pp. 45–53 (2014).
[3] Duboue, P. A.: Extractive email thread summarization: Can we do better than He said She said?, *Proceedings of INLG2012*, pp. 85–89 (2012).
[4] Oya, T. and Carenini, G.: Extractive Summarization and Dialogue Act Modeling on Email Threads: An Integrated Probalistic Approach, *Proceedings of SIGDIAL2014*, pp. 133–140 (2014).
[5] Mihalcea, R. and Tarau, P.: TextRank: Bridging Order into Texts, *Proceedings of EMNLP2004*, pp. 404–411 (2004).
[6] Rush, A. M., Chopra, S. and Weston, J.: A Neural Attention Model for Sentence Summarization, *Proceedings of EMNLP2015*, pp. 379–389 (2015).
[7] Kikuchi, Y., Neubig, G., Sasano, R., Takamura, H. and Okumura, M.: Controlling Output Length in Neural Encoder-Decoders, *Proceedings of EMNLP2016*, pp. 1328–1338 (2016).
[8] Luong, M.-T., Pham, H. and Manning, C. D.: Effective Approaches to Attention-based Neural Machine Translation, *Proceedings of EMNLP2015*, pp. 1412–1421 (2015).
[9] Cao, Z., Luo, C., Li, W. and Li, S.: Joint Copying and Restricted Generation for Paraphrase, *Proceedings of AAAI17*, pp. 3152–3158 (2017).
[10] Hochreiter, S. and Schmidhuber, J.: Long Short-Term Memory, *Neural Computation*, Vol. 9, No. 8, pp. 1735–1780 (1997).
[11] Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. and Bengio, Y.: Learning Phrase Representations using rnn encoder-decoder for statistical machine translation, *Proceedings of EMNLP2014*, pp. 1724–1734 (2014).
[12] Lin, C.-Y.: ROUGE: A Package for Automatic Evaluation of Summaries, *Proceedings of ACL2004 Workshop*, pp. 74–81 (2004).
[13] Bahdanau, D., Cho, K. and Bengio, Y.: Neural Machine Translation by Jointly Learning to Align and Translate, *Proceedings of ICLR2015* (2015).