

レビューからの質問文生成のためのILP文抽出

濱下 昌克† 乾 孝司† 村上 浩司§ 新里 圭司§

† 筑波大学大学院システム情報工学研究科 § 楽天技術研究所

1 はじめに

近年、ユーザが SNS などへ投稿した製品レビューの分析結果を製品開発へフィードバックすることが可能な環境が整ってきた。しかし、レビューの中には一言だけの短い書き込みも多く、当然ながら、このような書き込みから得られる情報は非常に少なくなる。この状況を改善するアプローチとして、ユーザからのレビュー投稿を受動的に待つのではなく、能動的に問いかけることでユーザ意見を収集する事が考えられる。そして、この中核に質問文生成課題がある [1]。

ここで言う質問文生成課題とは、問いかけ内容をあらかず単語や文が入力として与えられた時、入力情報から問いかけ用の質問文を生成する課題である。従来からあるテンプレートに基づく素朴な質問文生成手法では「○○は□□でしたか?」のようなテンプレートを事前に用意しておき、問いかけ内容をあらかず単語をテンプレート中のスロットに埋めることで質問を生成する。しかし、その表現力はテンプレート数に依存し、一般に、テンプレートに基づく手法では表現力が乏しいという問題がある。

本研究では、この問題を解決するために、過去に蓄積されたレビュー集合を豊富な表現リポジトリとして活用した、レビュー要約に基づく質問文生成手法を提案する。同じ製品 (例えば、ある銘柄のワイン) を購入したユーザであっても感想はさまざまであり (例 1)、このような感想を元に質問文に生成する (例 2) ことで、テンプレートでは実現が困難な、表現力豊かなさまざまな質問が生成できると期待できる¹。

- 例 1 : 軽めだと思う / さわやかな味わいのワインでした / 味わいはまるで草原のよう
- 例 2 : 軽めだと思いますか? / さわやかな味わいのワインでしたか? / 味わいはまるで草原のよう?

ただし、レビュー集合に含まれる文の中には質問文生成に適当でない文も多くある。そこで、レビュー集合から質問文生成に適当な文を選ぶための文抽出型レビュー要約手法を新たに提案する (詳細は 4 節で述べる)。

¹もちろんであるが、意見収集の際は、ある感想を元に生成した質問文は、その感想の書き手とは異なるユーザに対して用いることを想定している。生成した質問文の利用は対話制御の話題であり、本稿では議論しない。

Step.1 の入出力例 (ワインレビュー)

0,s1, いつもありがとうございます。
 1,s2, さわやかな味わいのワインでした。
 0,s3, 特にブルーベリーの果実香が感じられ、軽快な酸のバランスが非常に取れており、おもしろい。
 1,s4, またリピートしようと思います。
 0,s5, そういえば去年とボトルが違っていましたね。

図 1: Step.1 の入出力例

2 質問文生成

本節では、文抽出型レビュー要約手法の説明をおこなう前に質問文生成の概略を示すことで、文抽出の動機づけを改めて確認する。

質問文生成は次の 2 段階の過程を経て実行される。

● Step.1 : 文抽出

レビュー文集合から質問文生成に相応しい文を抽出する (文抽出型レビュー要約を実行する)。

● Step.2 : 質問文変換

先の Step.1 で抽出された文をそれぞれ質問文に変換する。

Step.1 でレビュー文集合から質問文生成に相応しい文を抽出する。図 1 は Step.1 の入出力例である。文頭の数字が “1” の文が出力として抽出すべき文の例である。レビューには、挨拶やお礼の言葉など、質問文生成に不要な文が含まれている (図 1 の s1)。また、ユーザの感想が引き出せる質問内容をもつが、複数話題が含まれ非常に長いなど Step.2 の質問文変換が困難な文 (s3)、逆に、質問変換が容易だとしても、作成した質問からユーザの感想が引き出せない文 (s5) もある。これらの文を取り除き、質問文生成に相応しい文のみ Step.2 へわたす。

Step.2 は質問変換に特化した換言処理である。既に、文法ルールによる単純な実装から大規模データを用いた機械学習手法 (例えば [2]) に至るまで汎用的に利用できる手法が存在する。そのため本稿では特に立ち入らない。

3 ILP 文抽出

文抽出課題は文書要約の一形態として以前から広く研究されている [3, 4]. このうち、整数線形計画問題 (Integer Linear Programming; ILP) による文抽出手法は、柔軟なモデル拡張性をもち、多くの先行研究で採用されている [5, 6]. 本稿で提案する文抽出型レビュー要約手法も ILP 文抽出手法であるため、本節でまず ILP 文抽出について説明する.

ILP 文抽出では、入力として文集合 $D = \{s_1, \dots, s_N\}$ が与えられ、そこから適切な部分集合 $S \subseteq D$ を構成する問題として捉える. ここで、入力文集合を N 次元の 0/1 ベクトル $\mathbf{y} = \{y_1, \dots, y_N\}$ であらわすと、 D 中のある文 s_i が $s_i \in S$ ならば $y_i = 1$, そうでないなら $y_i = 0$ となるベクトルとして文抽出の結果をあらわすことができる.

ILP 文抽出の基本モデルは次のようになる.

$$\begin{aligned} \mathbf{y}^* &= \arg \max_{\mathbf{y}} f(\mathbf{y}) \\ \text{s.t.} \quad &\sum_{i=1}^N l_i y_i \leq L_{max} \\ &\forall i, \quad y_i \in \{0, 1\} \end{aligned}$$

関数 $f(\mathbf{y})$ は出力候補 \mathbf{y} の良さを測る評価関数であり、制約式を満たしつつ、この値が最大となる候補を出力する. ここで、制約式内の L_{max} は出力の最大長、 l_i は文 s_i の長さをあらわしている. ILP 文抽出では、 $f(\mathbf{y})$ の定義および制約式の追加によって目的に合った文抽出器を設計する.

4 提案手法

本研究が文抽出器にもとめる要件は以下である.

- 意見を含む文を優先的に抽出する.
- 構文的、意味的に簡潔な文を優先的に抽出する.
- 出力される文集合全体としてさまざまな意見を含まう文を抽出する.

要件 a) によって図 1 の s1 や s5 のような文を排除し、要件 b) によって s3 のような文を排除する. また、要件 c) によって表現力を確保する.

以上の要件をすべて満たす文抽出モデルとして、評価関数を次のように定義し、

$$f(\mathbf{y}) = \sum_{j=1}^{|Q_a|} \sum_{k=1}^{|Q_e|} \frac{b_{jk}}{d_{jk}} z_{jk} \quad (1)$$

この評価関数を使って基本モデルを次のように拡張する.

$$\begin{aligned} \mathbf{y}^* &= \arg \max_{\mathbf{y}} f(\mathbf{y}) \\ \text{s.t.} \quad &\sum_{i=1}^N l_i y_i \leq L_{max} \\ &\forall i, \quad \sum_{j=1}^{|Q_a|} c_a(\mathbf{y}_i, a_j) = \sum_{k=1}^{|Q_e|} c_e(\mathbf{y}_i, e_k) \quad (2) \end{aligned}$$

$$\forall i, \quad \sum_{j=1}^{|Q_a|} c_a(\mathbf{y}_i, a_j) \leq 1 \quad (3)$$

$$\forall i, \quad \sum_{k=1}^{|Q_e|} c_e(\mathbf{y}_i, e_k) \leq 1 \quad (4)$$

$$\forall j, k, \quad z_{jk} \leq \sum_{i=1}^N o_{ijk} y_i \quad (5)$$

$$\forall i, \quad y_i \in \{0, 1\}$$

$$\forall j, k, \quad z_{jk} \in \{0, 1\}$$

式 (1) の評価関数について説明する. 本研究では意見を評価視点 (aspect) と評価表現 (sentiment) のペアとして定義し、評価視点 (a_j) の集合を Q_a , 評価表現 (e_k) の集合を Q_e とする. 変数 z_{jk} は、意見 $\langle a_j, e_k \rangle$ が出力に含まれるなら 1, 含まれないなら 0 の値をとる. これをすべての意見で足し込む ($\sum_{j=1}^{|Q_a|} \sum_{k=1}^{|Q_e|} \frac{b_{jk}}{d_{jk}} z_{jk}$) ことで、さまざまな意見を含む程、評価関数の値が大きくなる. ここで、 $\frac{b_{jk}}{d_{jk}}$ は意見 $\langle a_j, e_k \rangle$ に対する重みをあらわし、分子 b_{jk} はレビューにおける $\langle a_j, e_k \rangle$ の生起確率であり、分母 d_{jk} は意見の構成要素である評価視点 a_j と評価表現 e_k 間のレビュー内における平均係り受け距離である. この重み付けによって、よく使われる標準的な言い回しで、かつ、評価視点と評価表現が近接して現れる (すなわち構文的に簡潔な) 意見が優先的に抽出される.

次に、基本モデルから追加された制約式について説明する. 制約式 (2) 内の関数 $c_a(\mathbf{y}, a_j)$ は、

$$\sum_{i=1}^N h_{ij} y_i$$

と定義され、 h_{ij} は文 s_i に評価視点 a_j が含まれていれば 1, そうでなければ 0 をとる. つまり、 $c_a(\mathbf{y}, a_j)$ は a_j を含む文が出力される数を示す. ここで、ベクトル \mathbf{y}_i は \mathbf{y} の y_i 要素以外の値を 0 にしたベクトルである. 結果として、 $c_a(\mathbf{y}_i, a_j)$ は s_i に a_j が含まれていれば 1, そうでなければ 0 をとり、 $\sum_{j=1}^{|Q_a|} c_a(\mathbf{y}_i, a_j)$ は s_i に含まれる評価視点の異なり数をあらわす. 同様に $\sum_{k=1}^{|Q_e|} c_e(\mathbf{y}_i, e_k)$ は s_i に含まれる評価表現の異なり数をあらわす. 以上をまとめると、制約式 (2) は s_i が出力に含まれるには、 s_i に含まれる評価視点の異な

り数と評価表現の異なり数が同数である（評価視点と評価表現がペアとして出現し意見を構成する）ことを課している。制約式 (3) および (4) はその数がたかだか 1 つであることを課している。この 2 つの制約は、意見を含む文を抽出する際、その意見は簡潔に表現されていることを要請する。制約式 (5) は、文 s_i の出力に関する変数 y_i と出力に含まれる意見に関する変数 z_{jk} 間の整合性を保つための関係式である。この式の o_{ijk} は文 s_i に意見 $\langle a_j, e_k \rangle$ が含まれていれば 1、そうでなければ 0 をとる。つまり、出力中に意見 $\langle a_j, e_k \rangle$ が含まれるためには、 $\langle a_j, e_k \rangle$ を含む文が 1 つ以上出力されなければならない事を課している。

5 評価実験

5.1 実験設定

入力データとして、楽天市場でワインを扱う店舗に書かれたレビューに含まれる文集合のデータを使用した。元レビューデータに対し、2 名によるアノテーション作業の結果、抽出する（正例）／しない（負例）の判断が一致していた文の集合を選択して実験に用いた。データの詳細を表 1 に示す。

表 1: 入力データの詳細

文数 (正例/負例)	Kappa 値
715 (367/348)	0.765

意見の重み（意見の生起確率） b_{jk} は、入力データとは別に用意した大規模レビューを用いて算出した。評価視点集合 Q_a は各ドメインに対して人手で約 200 件の単語を選択して準備した。濱下らの調査 [7] によると、否定的な文言は質問文に用いない方がよい。そこで、評価表現集合 Q_e の要素には Positive 極性をもつ評価表現のみおよそ 700 語を採用した。最大出力長 L_{max} は、要約率 5% となるよう設定した。提案手法の ILP の解は Python の Pulp ライブラリ [8] を用いて得た。

5.2 実験結果

実験結果を表 2 に示す。また、出力例を図 2 に示す。今回はモデル比較用に、出力文内に含まれる意見を制御する制約式 (2) と、制約式 (3) および (4) ((3) と (4) は 1 対で使用する制約式) の使用有無を変更した 4 つのモデルを試した。表 2 の左列がモデルの違いを示しており、たとえば、“T/T” は式 (2) を使用し、かつ式 (3)(4) も使用すること、“T/F” は式 (2) は使用するが、式 (3)(4) は使用しないことをあらわす。

図 2 から、要件 a) を満たす、意見を含む文を抽出できていることがわかる。次に、各モデルの違いを表 2

表 2: 文抽出実験の結果

	抽出 文数	抽出 適合率	異なり 意見数	文字数 /文
T/T	14	0.857	14	24.8
T/F	22	0.727	48	38.3
F/T	14	0.857	14	24.8
F/F	45	0.556	146	42.5

を見ながら考察する。まず、T/T モデルと F/F モデルで大きく出力結果の特徴が異なることがわかる。T/F モデルはその中間であり、F/T モデルは今回の設定では T/T モデルと同じ結果であった。そこで、T/T モデルと F/F モデルの結果の詳細を比較する。まず、抽出適合率を見ると、T/T モデルが高い値を示しており、質問文生成に相応しい文を F/F モデルよりもうまく抽出できていることがわかる。異なり意見数は F/F モデルの方が多いが、これは文あたり文字数からもわかるように、F/F モデルは長い文を選び、評価関数値を上げるために 1 文に複数の意見が含まれた文を優先的に出力した結果であると言える。これは要件 c) には合致するが、要件 b) から見れば適した結果ではない。一方、T/T モデルは逆の傾向を示しており、制約式 (2)(3)(4) の作用により意見をひとつ含む短い文を優先的に出力している。T/T モデルは抽出文の文長が短いことから抽出文数が F/F モデルよりも多くなる事が期待されるが、結果はそうになっていない。この理由は、T/T モデルがもつ制約を満たす文が今回のデータセットにわずかしこ含まれておらず、 L_{max} 限界近くまで文出力がなされなかったためである。入力に制約を満たす文が存在すれば、抽出文数が増え、それに伴って抽出される異なり意見数も延びると予想される。

6 関連研究

本研究は意見要約や意見抽出の研究と関連が深い、次のような違いがある。

文抽出型レビュー要約はレビュー集合から意見を含む文を抽出することで要約文書を作成することが目的であり、本研究と非常に近い。しかし、レビュー要約は文書出力を仮定するため、要約中の文の順序を考慮したモデル設計が必要になる [6]。一方で、本研究は文抽出型レビュー要約の枠組みから着想を得て提案モデルを設計しているが、出力はあくまで文集合であり、個々の文の内容に関心がある。そのため、提案モデルでは、文の順序は考慮せず、その代わりに文内の意見数を制御している。

意見抽出ではレビュー等の文書集合から網羅的に意見を抽出することが目的である [9]。この場合、抽出

文抽出実験の出力例

T/T,1,1, 長い余韻が楽しめる
T/T,1,1, 滑らかで豊かな味わい
T/T,0,1, 確かに香りだけではずっと優しい
T/T,1,0, 味は奥行きが深い
T/T,0,0, ワインにとっても重要な要素であるフレーバーは、ワインの果実味を引き出します
T/T,0,0, 夜間の低温がブドウの酸を保ちます
T/T,0,0, これらの設備を導入する目的は、技術に頼ったワインを造ることではなく、発酵温度の急上昇によって果実味を失わないよう、みずみずしい風味を最大限残すワイン造りをするためなのです
T/F,1,1, アタックからバランスの良い味わいである
T/F,1,1, 柔らかくてクリーム状のタンニンによって非常に良い丸みがある素晴らしいフルーティーな味わいです
F/F,1,1, イチゴやグレープフルーツの風味を発散する細かい泡とともに、まろやかで爽やかな味わい
F/F,1,1, 余韻の渋味は丸くスムーズなキレが楽しめます
F/F,1,1, ふくよかでバランスに優れた芳醇です
F/F,0,1, そんな、太陽・風・大地の恩恵を受けるプーリアで作られ、果実味たっぷりの、完熟した味わいが楽しめる『サン・マルツァーノ』のワイン
F/F,0,0, 畑を自然なバランスで保つ

図 2: 文抽出例. 左から, モデル, 正解 (抽出する 1/しない 0), 出力 (抽出した 1/しなかった 0), 対象文.

単位は文ではなく語句がしばしば採用されるため, 意見抽出では 4 節 で述べた要件 b) に関心がない.

7 おわりに

本稿では, 質問文生成の新しいアプローチとして, 過去に蓄積されたレビュー集合を表現リポジトリとして活用したレビュー要約に基づく手法を提案し, 特に ILP ベースの文抽出モデルに関する評価実験の結果を述べた.

今後の予定として, まず, 対象商品をワイン以外に増やした評価実験を実施することを通して, 提案モデルの特徴の違いをより詳しく調査することが挙げられる. また今回は文抽出ステップに焦点を当てた評価をおこなったが, 続く質問文変換ステップを接続した質問文生成全体の評価も進めたい.

参考文献

- [1] V. Rus, P. Piwek, S. Stoyanchev, B. Wyse, M. Lintean, and C. Moldovan. “Question Generation Shared Task and Evaluation Challenge: Status Report.” In Proceedings of the 13th European Workshop on Natural Language Generation, pp.318–320, 2011.
- [2] L. Nio and K. Murakami. “Intelligence is Asking the Right Question: A Study on Japanese Question Generation.” IEEE Spoken Language Technology conference, 2018.
- [3] J. Kupiec, J. Pedersen and F. Chen. “A Trainable Document Summarizer.” In Proceeding of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, pp.68–73, 1995.
- [4] T. Hirao, H. Isozaki, E. Maeda and Y. Matsumoto. “Extracting Important Sentences with Support Vector Machines.” In Proceedings of the 19th International Conference on Computational Linguistics, Volume 1, pp.1–7, 2002.
- [5] 西野正彬, 安田宜仁, 平尾努, 鈴木潤. クエリ中の語を含むことを保証するクエリフォーカス要約. 言語処理学会第 18 回年次大会, 2012.
- [6] H. Nishikawa, T. Hasegawa, Y. Matsuo, G. Kikui. “Opinion Summarization with Integer Linear Programming Formulation for Sentence Extraction and Ordering”. Proceedings of the 23rd International Conference on Computational Linguistics, pp.910–918, 2010.
- [7] 濱下昌克, 乾孝司, 村上浩司, 新里圭司. 対話型情報収集のための質問生成における否定辞挿入効果について. 言語処理学会第 24 回年次大会, 2018.
- [8] M. Stuart, M. OSullivan, and I. Dunning. “PuLP: a linear programming toolkit for python.” The University of Auckland, Auckland, New Zealand, http://www.optimization-online.org/DB_FILE/2011/09/3178.pdf, 2011.
- [9] B. Liu, M. Hu and J. Cheng. Opinion Observer: Analyzing and Comparing Opinions on the Web. In Proceedings of the 14th International World Wide Web Conference, pp.342–351, 2005.