

類似商品のレビューを用いた評価情報補完

藤井真梨乃¹ 山本岳洋² 湯本高行³

本研究では、新商品やマイナーな商品などのレビューが少ない商品に対して、類似する商品につけられたレビューを用いてレビューを補完する手法を提案する。たとえば、ユーザがあるテレビの新商品について調べているとき、「どれぐらいの場所をとるのか」というクエリに対して、大きさが類似する商品につけられたレビューを検索しユーザに提示するというものである。これにより、ユーザが商品購入を行う上で参考になる情報を増やすことができる。この手法を実現するために、本研究では大規模言語モデル（LLM）を用いる。具体的には、まずLLMを用いて類似商品などのレビューに現れる評価がどの属性について言及されているのかを抽出する。その後、ユーザが入力した調べたいことから検索中の商品の属性に関連するものを取り出し、その属性が類似する商品から属性について述べている評価を検索することで補完を行う。評価では、テレビを対象にいくつかのクエリにおいて、調べたいことと各商品属性との関連性判定の精度、および検索中の商品のレビューと提案手法で出力されたレビューの類似度を測定した。調べたいことと商品属性との関連性判定の精度は、 F_1 値が0.833という高い結果になった。また、検索している商品の実レビューとの類似度評価についても、ランダムで商品を選んだ場合より約1ポイント高いという結果になった。

1 はじめに

ECサイトで商品を購入する上で、レビューは購入の判断材料となる。インターネットの発展に伴ってオンラインで買い物をする人が増加しており、経済産業省の調査によると、物販系分野のBtoC-EC市場規模は2013年の5兆9,931億円から2022年の13兆9,997億円まで年々拡大している[18]。

しかし、新商品やマイナーな商品など、レビューが少ない商品では、ユーザは知りたい情報を満足に得られないという問題がある。たとえばAという新製品のテレビについている、画面の大きさに関するレビューを調べたかったとする。このとき、画面の大きさに関するレビューを検索しても、それについて述べられているレビューが少ない、もしくは存在しないことがある。

そこで、この問題を解決するために、大規模言語モデル（LLM）を用いてレビューを補完する手法を提案する。まずユーザは調べたい商品を選択する。次に、「どれぐらいの場所を取るか」といった調べたいことを文章で入力し、その文章からLLMを

用いて商品説明のスペック情報に存在する「サイズ」といった属性を選択する。そして、選択した属性の値が類似する商品から、「6畳和室に置いていますが、ジャストサイズ!」といった属性について述べているレビューをLLMを用いて取得し、ユーザに返答するという手法である。これにより、ユーザは商品についているレビューが少なくとも、商品に関する評価をある程度予測することができる。たとえば、テレビAと同じ画面サイズを持つテレビBに「10畳の部屋にぴったりな大きさだと思います!」といったレビューが付いていた場合、これはテレビAでも同様であるといえる。

このような、類似する商品のレビューを用いて検索している商品の評価を補完する研究は以前から行われている[2][8]が、LLMを用いたものは存在しない。LLMを用いることによって、ユーザによる入力やシステムによる返答の自由度や精度が向上すると考えた。

最終目標は、調べたい属性について述べている類似商品のレビューを基に、評価情報を生成することである。これを実現するため、本研究では評価情報生成の際に必要となる、調べたい属性について述べている類似商品のレビューを取得することを目的とする。

評価では、2つの観点で評価を行った。1つ目は、ユーザが調べたいことと属性の関連性判定の精度評価である。著者自身とLLMのそれぞれで関連性判定を行った結果から適合率、再現率、 F_1 値を算出した。この結果、 F_1 値が0.833となり精度は高いといえるが、適合率は0.758となり、0.926であった再現率より少し低いという結果になった。2つ目は、提案手法で取得した類似商品のレビューと検索中の商品のレビューとの類似度評価である。提案手法で取得した類似商品のレビューと、ランダムに取得した商品のレビューのそれぞれにおいて、検索中の商品とのレビューとの類似度を算出した。その結果、ランダムで取得した商品レビューよりも、提案手法を用いて取得した類似商品のレビューの方が約1ポイント類似度が高くなかった。

2 関連研究

2.1 商品レビュー分析

商品レビューから属性とその意見を抽出する研究が行われている。Huら[6]は、商品の属性を抽出し、その特徴に対して肯定的、否定的な意見をWordNetを用いて収集した。杉木ら[16]は、自然言語表現で入力されたクエリと商品レビューに対し、感性表現ソーラスを用いて（項目、値）の組を抽出した。Poriaら[9]は、埋め込み表現を用いてレビュー中で述べられている観点を抽出した。中野ら[14]は、SVMを用いて属性および意見が含まれているかを判定する分類器を作成し、レビューから属性-意見のペアを抽出した。Wangら[12]は、多層結合アテンション機構を提案し、属性とその意見を抽出した。佐藤ら[15]は、BERTを用いて商品説明文1文ごとに関連するレビューを分類した。Xuら[13]は、LLMを用いて観光レビューの要約を生成し、レビューとその要約を双方のLSTMとGRUに入力することで、観点のカテゴリ、観点、意見、感情極性を抽出した。

これらの研究では、商品レビューから属性や観点を抽出する

¹ 学生会員 兵庫県立大学大学院情報科学研究科
ad24f053@guh.u-hyogo.ac.jp

² 正会員 兵庫県立大学大学院情報科学研究科
t.yamamoto@sis.u-hyogo.ac.jp

³ 正会員 兵庫県立大学大学院情報科学研究科
yumoto@sis.u-hyogo.ac.jp

いう部分に重きを置いている。本研究では、LLMによって商品レビューから抽出された属性と意見を用いて、類似する商品の評価を補完する。

2.2 類似商品のレビューから評価を取得する研究

類似商品のレビューから評価を取得する手法の研究が行われている。Parkら[8]は、クエリ尤度モデルを用いて、調べたい属性が類似している商品から属性に関するレビューを取得した。Duら[2]は、有用度や網羅性も考慮して、調べたい属性が類似している商品から属性に関するレビューを取得した。Rozenら[10]は、ある商品のレビューには回答が書いていない質問に対して、RoBERTaで類似商品と質問を考慮できるモデルを構築し、別の商品の回答を基に補完する手法を提案した。

これらの研究では、入力が属性やYes/Noの二値で答えられる質問など、ある程度限られた入出力を行っている。本研究は、これらの研究の手法を踏襲しつつ、最終的には取得した類似商品のレビューを基に評価情報を生成する。

3 問題定義

本節では、本研究で取り組む問題の定義を行う。ある商品ジャンルは k 個の商品 $\{P^{(1)}, P^{(2)}, \dots, P^{(k)}\}$ からなる。商品はその仕様を伝える商品属性を持っており、属性名を a_x 、属性値を v_x とすると (a_x, v_x) と表される。たとえばテレビの場合、(画面サイズ、32型)や(重量、4.5kg)と表される。あるジャンルの商品が n 個の商品属性を持っているとき、商品 $P^{(i)}$ の属性集合は式(1)のように表すことができる。

$$P^{(i)} = \{(a_1^{(i)}, v_1^{(i)}), (a_2^{(i)}, v_2^{(i)}), \dots, (a_n^{(i)}, v_n^{(i)})\} \quad (1 \leq i \leq k) \quad (1)$$

ユーザはシステムに対して調べたい商品 $P^{(i)}$ と調べたいこと q を入力する。このとき、ユーザが調べたい商品 $P^{(i)}$ を商品クエリ、調べたいこと q をテキストクエリと呼ぶ。たとえば、テキストクエリの例として「どれぐらいの場所を取るか」「持ち運びやすさ」などがあげられる。

システムには商品集合 $\{P^{(1)}, P^{(2)}, \dots, P^{(k)}\}$ 、商品クエリ $P^{(i)}$ とテキストクエリ q が与えられる。このとき、システムはテキストクエリ q と関連する属性を選択する。このテキストクエリと関連する属性のことを属性クエリと呼ぶ。たとえば、「どれぐらいの場所を取るか」というテキストクエリに関しては「画面サイズ」「幅×高さ×奥行」が属性クエリとなる。この選択された m 個の属性クエリ集合を $A_q^{(i)} = \{(a_l^{(i)}, v_l^{(i)})\}_{l=1}^m \subseteq P^{(i)}$ と表す。そして、属性クエリ集合 $A_q^{(i)}$ に含まれる属性 $p_l^{(i)} = (a_l^{(i)}, v_l^{(i)}) \in A_q^{(i)}$ について、他の商品との類似度 $\text{sim}(p_l^{(i)}, p_l^{(j)})$ を求める。商品クエリと類似する商品 k 件 $S_q = \{P^{(1)}, \dots, P^{(k)}\}$ を求める。この S_q を「類似商品」と呼ぶ。類似商品 S_q の各商品に付けられたレビューを用いて、元の商品クエリに対して付与される可能性が高いレビューをランキングしユーザに提示する。

4 類似商品レビューを用いた評価情報補完

本節では、まず提案手法の概要について述べ、その後具体的な手法の説明を述べる。

4.1 提案手法の概要

提案手法の概要を図1に示す。まず、ユーザは商品クエリとテキストクエリを入力する。次に、そのテキストクエリから属性クエリをLLMを用いて取得する。商品クエリとそれ以外の商品で属性クエリの値の類似度を計算し、類似商品を求める。そして、類似商品のレビューから、属性クエリと関連している記述を取得する。属性-商品レビュー間の関連性判定は、すべての属性に対してあらかじめ行っておく。最後に、類似商品のレビューの中で属性クエリに関連すると判定されたものを検索結果に表示する。

4.2 属性クエリ判定

テキストクエリと関連する属性を、LLMを用いて判定する。本研究では、テキストクエリを商品クエリの機能に関する文章と限定する。機能に限定する理由は、商品の持つ属性には機能に関するものが多く、その情報を基に補完を行う必要があるからである。値段もレビューを読む上で重要な要素であるが、購入するタイミングによって変動するため、本研究では除外する。本判定において、プロンプトは以下の通りである。

あなたは今[商品クエリのジャンル]のレビューを検索しようとしています。

「[テキストクエリ]」という入力に対して、「[属性クエリの候補]」という属性はあてはまりますか。あてはまるかをTrue/Falseで書いてください。

###OUTPUT_EXAMPLE:
True or False

例えばテレビについて検索する場合、{商品クエリのジャンル}には「テレビ」、{テキストクエリ}には「どれぐらいの場所を取るか」や「画質の良さ」、{属性クエリの候補}として「画面サイズ」や「画素数」などが入力される。出力がTrueであったものを属性クエリとする。

4.3 類似商品の選択

商品クエリにおける属性クエリの値とそれ以外の商品における属性クエリの値の間で類似度を計算し、類似商品を選択する。このとき、4.2節でTrueと判定された属性のみを属性クエリ集合 $A_q^{(i)}$ として使用して類似商品を選択する。ある商品 $P^{(i)}$ の属性 $p_l^{(i)} = (a_l^{(i)}, v_l^{(i)})$ について、商品 $P^{(j)}$ の対応する属性 $p_l^{(j)} = (a_l^{(j)}, v_l^{(j)})$ との類似度を求める。まず、属性が量的データの場合、以下の式に従って類似度を求める。

$$\text{sim}(p_l^{(i)}, p_l^{(j)}) = 1 - |v_l^{(i)} - v_l^{(j)}| \quad (2)$$

ここで、 $v_x^{(i)}$ は最大値が1となるように正規化された値である。また、属性がカテゴリカルデータである場合、以下の式に従って属性値が完全に一致する場合にのみ類似度を求める。

$$\text{sim}(p_l^{(i)}, p_l^{(j)}) = \begin{cases} 1 & \text{if } v_l^{(i)} = v_l^{(j)} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

本研究では、属性クエリ集合 $A_q^{(i)}$ の各属性についてこれらの式で類似度を求め、類似度の合計値が最も高い商品から上位 N 件の商

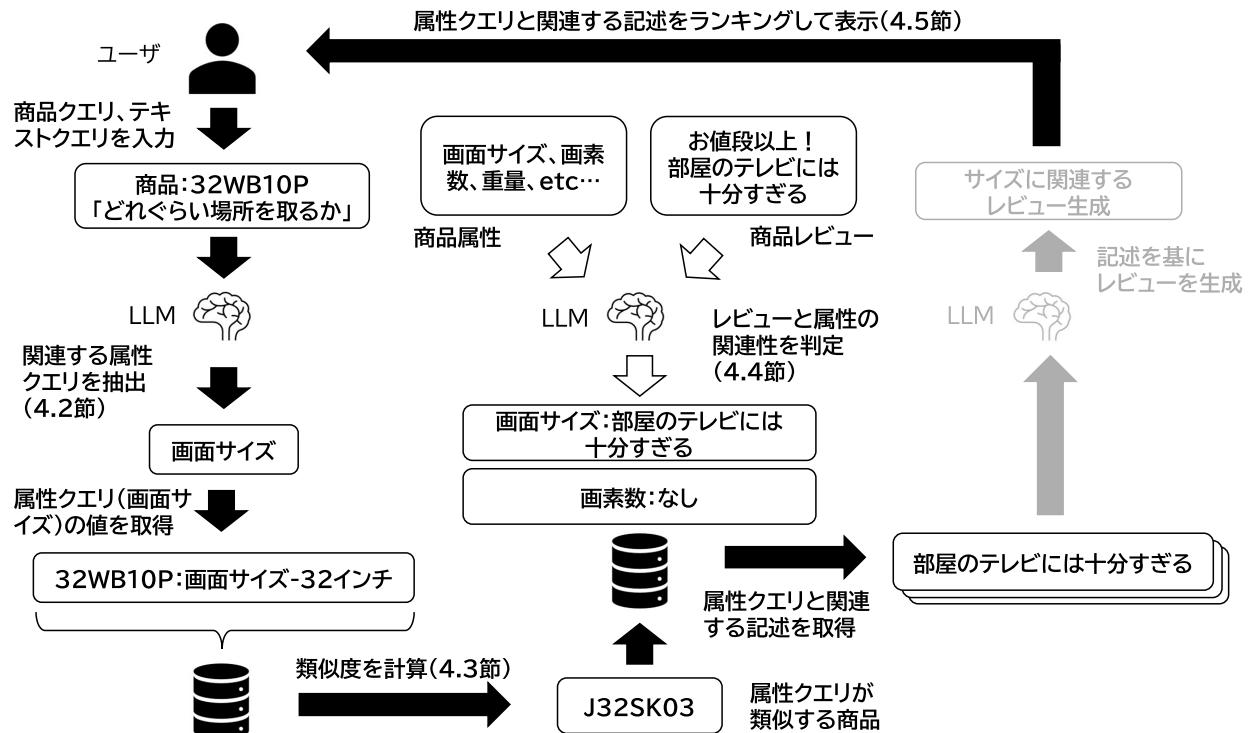


図1 提案手法の概要.

品を類似商品として用いる。類似度が同値だった場合はレビュー数が多い商品を上位とする。

4.4 商品レビューと属性の関連性判定

商品レビューの中で属性について述べている部分をLLMを用いて抽出する。属性について述べている部分を抽出し、述べていない属性については「なし」と出力することで、関連があるかを判定することができる。

プロンプトでは、属性クエリとレビューを入力し、属性クエリに関連する記述を抽出するように指示を与える。精度の高い回答を生成するために、few-shotプロンプトを用いて例も与える。このとき、この例が出力に影響を与えないようにするために、現在調べているジャンルとは異なるジャンルの属性とレビューを使用する。テレビのレビューを用いる場合、プロンプト例は以下の通りである。今回はfew-shotの例として、ドライバーのレビューを与えている。

```

テレビのレビュー「{テレビジャンルの商品のレビュー}」から
「{属性クエリの1つ}」に関連する記述を抜き出してください。
OUTPUT_EXAMPLEの形式に従って出力してください。
関連する記述が無い場合、attributeの値は「{属性クエリの1つ}」、
sentenceの値は「なし」と出力してください。
{jsonで出力するための指示}
###INPUT_EXAMPLE:
(ドライバーのレビューの場合)
このドライバー、パワーが強くてすぐ乾きます。
###OUTPUT_EXAMPLE:
ドライバーのレビューから風量に関連する記述を抜き出す場合
{{attribute:風量, sentence:パワーが強くてすぐ乾く}}
INPUT:{商品クエリのジャンルの商品のレビュー}
JSON_OUTPUT:

```

たとえば、テレビのレビュー「デカくて安くて綺麗で最高!!!! 映画もこのでっかいテレビで見れば映画館行かなくてもいいかもね！」から「画面サイズ」や「録画機能」について抽出を行うとする。このとき、上記プロンプトの{テレビジャンルの商品のレビュー}には「デカくて安くて綺麗で最高!!!! 映画もこのでっかいテレビで見れば映画館行かなくてもいいかもね！」、{属性クエリの1つ}には「画面サイズ」や「録画機能」が入る。プロンプトを実行して「画面サイズ」に関連する記述を抜き出すと、「デカ

表1 用意した属性一覧。

属性	
画面サイズ	画素数
パネル種類	HDR方式
映像処理エンジン	バックライト
量子ドット	倍速機能
BS 8K	BS 4K/110度CS 4K
録画機能	ドライブ内蔵
自動録画機能	2番組同時録画
早見再生	回転式スタンド
スピーカー数	幅x高さx奥行(数値)
重量(数値)	リモコン(音声操作)
スマートスピーカー連携	HDMI端子

表2 用意したクエリ。

テレビ	画質の良さ
	どれぐらいの場所を取るか
	音質の良さ
	持ち運びやすさ
	沢山録画できるか
	リモコンが使いやすいか
	スマートスピーカーとちゃんと接続できるか
	回転式スタンドの動かしやすさ
	映画鑑賞に向いているか
	ゲーム機との相性の良さ

くて」という記述が出力される。また上記のレビューには「録画機能」という属性については関連する記述が存在しないため、この属性をプロンプトに入力して実行すると、出力は「なし」となる。

4.5 属性クエリに関するレビュー中の記述のランキング

4.4節で取得した、類似商品のレビュー中に存在する属性クエリに関する記述をユーザに表示する。このとき、表示する記述については、2通りの手法でランキングした後、それぞれのランキングを結合する。

まず、それぞれのランキング手法について述べる。

1つ目は、LexRank [3]を用いる手法である。LexRankは文集合の中から重要な文を発見するために用いられる手法の1つである。LexRankでは、文間のコサイン類似度を重みとする隣接行列で表し、その隣接行列の最大の固有値に対応する固有ベクトルを求め、固有ベクトル中の値の降順で並び替えることによって文のランクを行っている。LexRankを用いて、取得した記述の相対的な重要度をランキングすることにより、属性クエリについて多くのユーザが評価していることを把握することができる。

2つ目は、テキストクエリとのコサイン類似度を用いる手法である。この手法を選んだ理由は、多くのユーザが述べていることを取得しても、それがユーザの知りたいことであるとは限らないため、ユーザの入力を考慮したランキングも必要であるからである。この手法についても、コサイン類似度を降順で並び替えた結果をランキング結果として使用する。

両方の手法において、文やテキストクエリとのコサイン類似度を求めるときにはembeddingモデルを用いて文をベクトル化する。

そして、ランキングの結合にはRRF [1]という手法を用いる。RRFとは、各手法の検索結果の逆順位を合計し、そのスコアの降順でランキングしなおす手法である。属性に関連する記述 d のLexRankにおける順位が $\text{rank}_1(d)$ 位、テキストクエリとのコサイン類似度による順位が $\text{rank}_2(d)$ 位であるとき、スコアの

算出方法を式(4)に示す。

$$\text{RRF}(d) = \frac{1}{k + \text{rank}_1(d)} + \frac{1}{k + \text{rank}_2(d)} \quad (4)$$

k は定数であり、今回は一般的に使用される値である60とした。

最後に、実際の出力例について述べる。「32WB10P」という型番のテレビを商品クエリとして、「どれぐらいの場所を取るか」というテキストクエリを入力したとき、LexRank、クエリとの類似度およびRRFでランキングして上位5件に表示される記述を表3に示す。「32インチはちょうど良いです」や「10畳の部屋にぴったりな大きさだと思います!」というように、32型のテレビについて、大きさに関するレビュー中の記述を取得することができていることが分かる。

5 実験

5.1 使用データ

本節では実験で使用するデータについて説明する。商品属性および属性値のデータは価格.com^{*1}のスペック情報を使用する。このデータを使用した理由は、同じジャンルの商品であればスペック情報の項目はまったく同じであるためである。具体的には、「液晶テレビ・有機ELテレビ」ジャンルの項目の中から、表1の項目を使用した。

商品レビューについては楽天データセットに含まれる楽天市場の商品レビューデータ [17]を利用した。具体的には、テレビジャンルの中からレビュー数の上位100件を取得し、そのうち型番が明記されており、かつ価格.com上にスペック情報が掲載されている54件の商品、計4,071件のレビューを使用する。

5.2 属性クエリ判定の精度評価

4.2節ではLLMを用いてテキストクエリが属性と関連するかについて判定を行っている。どの程度正しく関連性を判定できているかを調べるために、精度評価を行った。

^{*1} <https://kakaku.com/>

表3 「どれぐらい場所を取るか」と入力したときの出力例。

LexRank	クエリとの類似度
1. 32インチ大きい画面	1. 設置スペースがこのサイズがいっぱいなのでこれにしました。
2. 32インチはちょうど良いです	2. 10畳の部屋にぴったりな大きさだと思います!
3. 同じ画面サイズ32インチを選択しました	3. ちょうど良いサイズ
4. この値段でこの画面の大きさ画質に満足しています	4. この大きさ。
5. 大画面でいいですね	5. 画面の大きさ
RRFで合算	
1. 設置スペースがこのサイズがいっぱいなのでこれにしました。 2. 10畳の部屋にぴったりな大きさだと思います! 3. 32型で5Kgと軽い 4. 部屋は6畳ぐらいなので、ちょうどいい大きさ。 5. 画面の大きさ	

表4 テキストクエリと属性クエリの関連性判定の精度。

評価指標	評価値
適合率	0.758
再現率	0.926
F_1 値	0.833

5.2.1 実験方法

使用するクエリは表2の通りである。また、使用する属性は表1の通りである。LLMはgpt-4o-2024-08-06^{*2}を用いた。このとき、temperatureは0とした。

それぞれのクエリについて、属性がクエリに関連すると著者1人で判定した結果及びLLMで判定した結果を比較し、適合率、再現率、 F_1 値を算出した。

5.2.2 結果

評価結果を表4に示す。再現率が0.926と非常に高く、人手で関連すると判定された属性はLLMでもほとんど正しく判定されたことがわかる。そして、適合率は0.758と再現率と比べて低いことがわかる。

適合率が再現率より低くなった要因は、テキストクエリとあまり関係のない属性でも、関連すると判定したものがいくつか存在したことである。実際に、「リモコンが使いやすいか」というテキストクエリに対して「重量」「バックライト」が関連すると判定された。ただ、LLMによる誤りが提案手法全体に与える影響は少ないと考えられる。なぜなら、提案手法ではランキングを行うことで、クエリと類似度が高く、多くのユーザが言及している評価のみを取得しているからである。テキストクエリとあまり関係のない属性について言及している評価が、ランキング上位に来ることは少ないと考えられる。

5.3 商品クエリと類似商品のレビューの類似性評価

提案手法の有効性を検証するため、提案手法で取得したレビュ

ーと、商品クエリに実際についているレビューの類似度を算出した。この評価の目的は、商品クエリのレビューが存在しないと仮定したときに、提案手法で取得したレビューが商品クエリに対する評価情報の補完として適切かを調べることである。提案手法により類似商品から得られるレビューのランキングと、商品クエリに実際についているレビューのランキングが類似していれば、提案手法により商品クエリの評価情報を補完できると考えられる。

5.3.1 実験方法

商品クエリと類似する商品のレビューの中で、テキストクエリと関連する記述を提案手法を用いて取得し、実際に商品クエリの中の記述とのコサイン類似度の最大値を比較した。テキストクエリに関連する記述の量を確保するため、レビュジャンルの中で最もレビューが多い商品「32WB10P」を商品クエリとし、レビュー中から提案手法を用いて取得したテキストクエリに関連する記述を正解データとする。

使用するクエリは表2のうち、商品クエリに該当する機能がついていない「スマートスピーカーとちゃんと接続できるか」を除いたものである。使用する属性は表1の通りである。

提案手法を実行するにあたり、4.2節で述べた、ユーザの入力から関連属性を判定する操作には、5.2節と同様にgpt-4o-2024-08-06を用いる。4.3節で述べた、類似商品の選択については、最大10商品までを選択する。今回使用する属性のうち、量的データをもつ属性は「重量」「幅×高さ×奥行」である。「幅×高さ×奥行」については、3値の合計値で類似商品を選択した。4.4節で述べた、商品レビューから属性に関連する記述を抽出する操作にはLlama-3.1-Swallow-8B-Instruct-v0.2 [4] [5] [7]を用いた。temperatureを0.001とし、ほぼ常時同じ回答が出力されるようにした。また、4.5節で述べた、文書間のコサイン類似度を計算するとき、文章のベクトル化にmultilingual-e5-large^{*3} [11]を用いた。

図2に類似度の評価方法を示す。4.4、4.5節の手法を用いて、

^{*2} <https://platform.openai.com/docs/models#gpt-4o>

^{*3} <https://huggingface.co/intfloat/multilingual-e5-large>

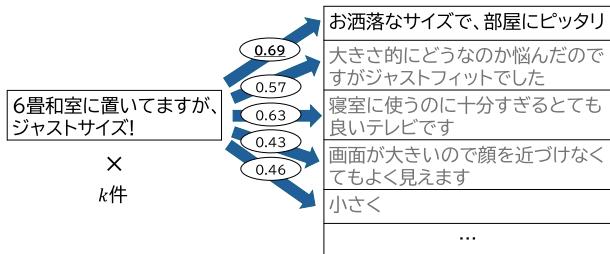


図2 類似性の評価手法。

表5 テキストクエリごとに取得した記述の件数。

テキストクエリ	商品クエリ	類似商品	ランダム
画質の良さ	220	1,078	273
どれぐらい場所を取るか	22	170	196
音質の良さ	40	93	31
持ち運びやすさ	311	258	229
沢山録画できるか	46	242	45
リモコンが使いやすいか	173	311	106
回転式スタンドの動かしやすさ	149	102	70
映画鑑賞に向いているか	330	1,326	420
ゲーム機との相性の良さ	38	139	33

商品クエリ、類似商品ともに上位 k 件まで属性に関する記述を取得する。類似商品の記述1件に対して、商品クエリのレビューに存在する記述 k 件とのコサイン類似度を1件ずつ求め、その類似度の最大値をレビューの類似度と定義する。これを類似商品の記述 k 件で行う。

本研究では2種類の比較を行った。

1. 商品クエリと類似商品の中から取得するレビューの記述を、上位10~30件まで10件ずつ変化させる。
2. 類似商品10商品から取得したランキング、同ジャンルからランダムに選択した10商品から取得したランキングに対して、それぞれ商品クエリのレビューに存在する記述との類似度を求める。

テキストクエリについて、商品クエリ、類似商品、ランダムに選択した商品で取得した記述の件数を表5に示す。

5.3.2 結果

商品クエリと類似商品の中から取得するレビューの記述を、上位10~30件まで10件ずつ変化させた結果を表6に示す。この結果から、RRFによるランキングは上位 k 件という区切りでは、テキストクエリによって商品クエリのレビューとの類似度の変化傾向が異なるということがわかる。これは、テキストクエリごとに取得した記述の件数が異なるため、両方上位 k 件で区切ると類似度の分布に差が生じることが要因であると考えられる。

類似商品とランダムに選んだ商品のレビューについて、それ

表6 上位 k 件ごとの結果。

テキストクエリ	$k=10$	$k=20$	$k=30$
画質の良さ	0.996	0.993	0.994
どれぐらい場所を取るか	0.940	0.944	0.948
音質の良さ	0.927	0.929	0.932
持ち運びやすさ	0.963	0.967	0.963
沢山録画できるか	0.969	0.960	0.958
リモコンが使いやすいか	0.933	0.934	0.940
回転式スタンドの動かしやすさ	0.960	0.960	0.961
映画鑑賞に向いているか	0.982	0.976	0.971
ゲーム機との相性の良さ	0.927	0.933	0.933
全体平均	0.955	0.955	0.956

表7 テキストクエリごとの結果。

テキストクエリ	提案手法	ランダム
画質の良さ	0.996	0.986
どれぐらい場所を取るか	0.940	0.933
音質の良さ	0.927	0.928
持ち運びやすさ	0.963	0.940
沢山録画できるか	0.969	0.939
リモコンが使いやすいか	0.933	0.939
回転式スタンドの動かしやすさ	0.960	0.950
映画鑑賞に向いているか	0.982	0.967
ゲーム機との相性の良さ	0.927	0.912
全体平均	0.955	0.944

ぞれ商品クエリのレビューと類似度を計算した結果を表7に示す。この結果から、ランダムに商品を取得したときの値0.944よりも類似度が0.955と約1ポイント高く、より類似する商品レビューを上位に表示することができていることがわかる。実際に、「どれぐらい場所を取るか」というテキストクエリに対する出力結果を上位5件まで表8、9に示す。これを見ると、ランダムで商品を取得したときには、第2位に「16インチでも狭い部屋では十分です」という商品クエリの画面の大きさと異なる商品のレビューが入っている。ランダムで選択した商品のレビューから関連するクエリを取得すると、商品とは類似しない属性値を持つ商品のレビューも上位にランキングされてしまうため、商品クエリのレビューとの類似度が低くなってしまったといえる。

6 まとめと今後の課題

本研究では、新商品やマイナーな商品などのレビューが少ない商品に対して、類似する商品につけられたレビューを用いてレビューを補完する手法を提案した。具体的には、まずユーザが商品と商品について調べたい文章を入力し、LLMを用いて文章から関連する属性を選択する。そして、関連する属性から類似商品を探し、LLMを用いて類似商品のレビューから関連する属性についての記述を取得するというものである。

テキストクエリと属性の関連性判定の精度や類似するレビューが取得できているかについて評価した。テキストクエリと属性の

表8 「どれぐらいの場所を取るか」と入力したときの出力例（ランダム）。

ランダム
1. 8畳くらいの部屋でやや大きめ
2. 16インチでも狭い部屋では十分です
3. ちょうどいい大きさ
4. 大きすぎず小さすぎず、ちょうど良かったです
5. 6畳の部屋で使用。ちょっと寝る前に見たいと思って欲しかったので、ちょうど良い大きさ。

表9 「どれぐらいの場所を取るか」と入力したときの出力例（類似商品）。

類似商品
1. 設置スペースがこのサイズがいっぱいなのでこれにしました。
2. 10畳の部屋にぴったりな大きさだと思います！
3. 32型で5Kgと軽い
4. 部屋は6畳ぐらいなので、ちょうどいい大きさ。
5. 画面の大きさ

関連性判定について、精度は高いといえるが、テキストクエリと関連しない属性でもいくつか関連があると判定するものが存在したという結果になった。また、商品クエリと類似するレビューを取得できているかについては、ランダムで同数の商品からレビューを取得するときよりも類似するレビューを取得できているという結果になった。

今後の課題を2点挙げる。1点目は、属性とレビューの関連性判定についての精度評価である。属性とテキストクエリの関連性判定の精度評価は行ったが、属性とレビューの関連性についてはまだ行っていない。そのため、属性とレビューについての精度評価もテキストクエリとの評価と同様の手法で行う。属性とレビューについては、4.4節で述べている通り、属性に関連する記述を抽出しているため、抽出精度についても評価を行う。また、本研究で行った評価も含めてユーザ実験を行い、ユーザ視点での精度も評価する。2点目は、調べたい属性について述べている類似商品のレビューを基に、評価情報を生成することである。本研究で実際に取得した記述には具体的な記述も存在するが、ランキングの下位には抽象的な記述が非常に多い。そのため、今後は具体的な評価情報を生成することで、よりユーザが知りたい情報を得られるようにすることを目指す。

謝辞

本研究はJSPS科学研究費助成事業JP25K03229, JP24K03228, JP21H03775による助成を受けたものです。また、本研究では国立情報学研究所のIDRデータセット提供サービスにより楽天グループ株式会社から提供を受けた「楽天データセット」(https://rit.rakuten.com/data_release/)を利用しました。ここに記して謝意を表します。

参考文献

- [1] Gordon V Cormack, Charles LA Clarke, and Stefan Buetzcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, pages 758–759, 2009.
- [2] Na-na Du, Liu-tong Xu, and Hai Huang. MSRM: A novel model to retrieve meaningful opinion sentences for new products. In *Proceedings of the 2017 International Conference on Mathematics, Modelling and Simulation Technologies and Applications*, pages 214–221, 2017.
- [3] Güneş Erkan and Dragomir R Radev. LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479, 2004.
- [4] Kazuki Fujii, Taishi Nakamura, Mengsay Loem, Hiroki Iida, Masanari Ohi, Kakeru Hattori, Hirai Shota, Sakae Mizuki, Rio Yokota, and Naoaki Okazaki. Continual pre-training for cross-lingual LLM adaptation: Enhancing Japanese language capabilities. In *Proceedings of the 1st Conference on Language Modeling*, 2024.
- [5] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, and et al. The llama 3 herd of models, 2024.
- [6] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177, 2004.
- [7] Naoaki Okazaki, Kakeru Hattori, Hirai Shota, Hiroki Iida, Masanari Ohi, Kazuki Fujii, Taishi Nakamura, Mengsay Loem, Rio Yokota, and Sakae Mizuki. Building a large Japanese web corpus for large language models. In *Proceedings of the 1st Conference on Language Modeling*, 2024.
- [8] Dae Hoon Park, Hyun Duk Kim, ChengXiang Zhai, and Lifan Guo. Retrieval of relevant opinion sentences for new products. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 393–402, 2015.
- [9] Soujanya Poria, Erik Cambria, and Alexander Gelbukh. Aspect extraction for opinion mining with a deep convolutional neural network. *Knowledge-Based Systems*, 108:42–49, 2016.
- [10] Ohad Rozen, David Carmel, Avihai Mejer, Vitaly Mirkis, and Yftah Ziser. Answering product-questions by utilizing questions from other contextually similar products. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 242–253, 2021.
- [11] Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Multilingual E5 text embeddings: A technical report. *arXiv preprint arXiv:2402.05672*, 2024.
- [12] Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, pages 3316–3322, 2017.
- [13] Chun Xu, Mengmeng Wang, Yan Ren, and Shaolin Zhu. Enhancing aspect-based sentiment analysis in tourism using large language models and positional information. *arXiv preprint arXiv:2409.14997*, 2024.
- [14] 中野裕介, 湯本高行, 新居学, and 上浦尚武. 機械学習による商品レビューの属性-意見ペアの抽出. 情報処理学会研究報告, 2015-DBS-162(14):1–8, 2015.
- [15] 佐藤風希 and 鈴木優. 異なる観点を考慮した商品説明に対応した商品レビューの抽出. In 第15回データ工学と情報マネジメントに関するフォーラム, 3a-6-2, 2023.
- [16] 杉木健二 and 松原茂樹. カスタマーレビューに基づく商品検索のための感性表現ソースの構築. In 言語処理学会第15回年次大会発表論文集, pages 781–784, 2009.
- [17] 楽天グループ株式会社. 楽天市場データ. 国立情報学研究所情報学研究データリポジトリ.(データセット), 2020. <https://doi.org/10.32130/idr.2.1>.
- [18] 経済産業省. 令和4年度電子商取引に関する市場調査報告書. <https://www.meti.go.jp/press/2023/08/20230831002/20230831002-1.pdf>. 2024年11月28日閲覧.