

商品属性とレビューの対応関係学習に基づく言語モデル構築

河田 友香[†] 山本 岳洋[†] 大島 裕明[†] 柳田 雄輝^{††} 加藤 誠^{†††}

藤田 澄男^{††††}

[†] 兵庫県立大学大学院 情報科学研究科 〒 651-2197 兵庫県神戸市西区学園西町 8-2-1

^{††} 筑波大学大学院 情報学学位プログラム 〒 305-8550 茨城県つくば市春日 1-2

^{†††} 筑波大学 図書館情報メディア系 〒 305-8550 茨城県つくば市春日 1-2

^{††††} ヤフー株式会社 〒 102-8282 東京都千代田区紀尾井町 1-3

E-mail: [†]ad221019@gsis.u-hyogo.ac.jp, [†]t.yamamoto@sis.u-hyogo.ac.jp, [†]ohshima@ai.u-hyogo.ac.jp,

^{††}tynagi2@klis.tsukuba.ac.jp, ^{†††}mpkato@acm.org, ^{††††}sufujita@yahoo-corp.jp

あらまし 本研究では、「画素数: 3,240 万, 撮像素子: フルサイズ, 重量: 800g」のような商品の属性と、「遊ぶ子どもの表情もくっきりと撮影できます」のような, 商品进行评估するレビューとの関連性を予測できる言語モデルを提案する. ここでの関連性の予測とは, 与えられた属性集合に対してそのレビューが付与される確率を表す. このようなモデルが実現できれば, 商品に詳しくないユーザでも, 商品の属性とその商品を実際に使ったときの評価との関係を知ることができる. このようなモデルを実現するため, 事前学習済みの BERT を使用し, ある商品に付与されたレビュー中のテキストが, その商品に対するレビューなのか対応関係を予測するようにモデルを学習する. また, 属性に含まれる数値情報の大きさをモデルが理解できるようにするための工夫についても提案する. 価格.com から抽出したカメラに関する 50 商品 11,067 件のレビューを用いて実験を行った結果, 量的データを含む属性値をカテゴリ化し, カテゴリを数値を用いて表現した手法が最も正解率が高いことが分かった.

キーワード レビュー, 言語モデル, BERT

1 はじめに

EC サイトで商品を購入する際に様々な情報を閲覧する. その中に商品の属性が挙げられ, それらを見ることは非常に重要である. 例えば, カメラだと「撮像素子」には「フルサイズ」や, 「画素数」には「3,410 万画素」というように, 「属性名」と「属性値」のペアの集合で属性を表すことができる.

内田らの研究では, 商品に人が触れることなく評価できる属性を探索的属性, 人が触れることで評価できる属性のことを経験的属性として定義している [20]. ここで, 「画素数」は「3,410 万画素」, 「撮像素子」は「フルサイズ」といった商品の属性名や属性値は, サイトなどの記述を見ることで理解することができるため, 探索的属性である. 「色鮮やかに画質良く撮影できる」や「軽く持ち運びに便利」のようなレビューや商品説明文は, 商品を使用することで初めてわかる評価であるため, 経験的属性である.

この経験的属性と探索的属性の関係を理解しながら, 商品を探すことは重要である. 例えばカメラを購入するときを考えてみる. 「子どもを撮影するので, 速い物を撮影できるカメラが欲しい」や「花や山など風景を撮影するので, 色鮮やかに画質良く撮影できるカメラが欲しい」のような経験的属性を考慮しながらカメラを探すときにはレビューや商品説明文を見るだろう. レビューや商品説明文には, 自分が求める経験的属性が記述されているものが確実に記述されているとは限らない. 例えば,

「色鮮やかに画質良く撮影できる」というレビューをどのユーザも書いていないカメラがあるとする. レビューのみから商品を探すとなると, その商品は, 色鮮やかに画質良く撮影できると捉えることは難しい. しかし, 商品の属性名, 属性値のような探索的属性に着目すると色鮮やかに画質良く撮影することができるカメラである可能性が考えられる. このようにカメラのことをあまり知らないユーザであれば, 自分が着目している経験的属性と関連する探索的属性を探すことは難しく, また, 現在の EC サイトではそのように商品を探すことはできない.

そこで本研究では, このような問題を解決するために, テキストである経験的属性と, 商品の属性名, 属性値である探索的属性の対応関係を学習し, 経験的属性と探索的属性の関連を理解した言語モデルを構築する. 商品のスペックをあらわす属性名と属性値の集合とテキストを与え, 属性の集合に対して, 入力したテキストと関連しているのかを判断する. このモデルにより, 属性名や属性値と関連している評価なのかを判断することができ, 探索的属性である商品の属性と経験的属性であるテキストの関連を理解することができる.

レビューと商品の属性名, 属性値の対応関係を理解した言語モデルを構築することで, レビューと商品の属性を関連させ, 商品を探すことができる. 例えば, 大量のレビューの中から, 「撮像素子」や「画素数」のように, 属性に関連するレビューを探すことができると考えられる. また, 「画質が美しく速い物体を撮影できるカメラ」のような経験的属性を入力とすると, その経験的属性が現れるような属性値を持つ商品を探し, その商

品を推薦することができる。このように属性名、属性値からレビューを探すことや、経験的属性から商品を推薦することなど多様な見方で商品を探すことが可能になる。

本研究では、レビューと商品の属性名、属性値の対応関係を学習させ、それぞれの関連を理解した言語モデルを構築する。ここで、数値を含む商品スペックのような表データとテキストデータであるレビューの対応をどのようにとるのが課題として挙げられる。そこで、本研究ではBERTを用いたレビューと商品の属性名、属性値の対応関係を学習させる。属性集合とレビューを入力とし、BERTを用いて属性集合とレビューがどの程度関連しているかを予測するモデルを構築する。このモデルを、正例を入力された属性集合を持つ商品に対して実際につけられたレビュー、負例を入力された属性集合を持つ商品とは異なる商品に対して実際につけられたレビューとし学習を行い、関連度合いを表すスコアを算出する。また、技術的課題である属性値の量的データの処理については、カテゴリ化や規模感を捉えるための処理をすることで言語モデルでの入力として適切に捉えることを可能にする。

2 関連研究

2.1 商品属性やレビューを用いたテキスト分析

商品にまつわるレビューのみ、また商品の属性やユーザの情報のみを用いたテキスト分析が行われている。Chenらは、ユーザの情報をもとにナレッジベースのニューラルネットワークモデルを構築し、ユーザの興味関心に合わせた商品説明文を生成する手法を提案している[1]。McAuleyらは、Yes/Noで判断できる質問に対して、レビューから回答を見つけ、答えを提示するモデルを提案している[12]。Royらは、商品の素材などの商品属性から回答できる質問に対して、商品属性から自動的に回答を生成するモデルをXLNetやBERTをもとに構築している[13]。Zhangらは、ユーザが行った商品に関する質問から、その質問に対する答えを予測し、答えに関するレビューを提示している[15]。Dongらは、ユーザの情報、評価レート、商品のIDを属性としてレビューを生成するモデルを構築している[5]。Dengらは、レビューから意見を集約し、回答を生成するモデルを提案している[3]。Chenらは、レビュー文から質問に関するスニペットを生成し、レビュー文に基づいて質問回答を生成するモデルRAGEを構築している[2]。

また、レビューと属性の両方を用いたテキスト分析が行われている。Fengらは、商品の属性とレビューの関連づけるモデルを構築し、属性とレビューの関係を理解した上で回答を生成する手法を提案している[6]。Gaoらは、商品属性と属性値、過去にユーザによってつけられたレビューを用いて、質問に関する属性に言及したレビューから自動的に回答を生成している[8][9]。Zhangらは、属性とBERTでベクトル化したテキストを結合するMA-Transformer Encodersを用いて、商品属性とテキストをベクトル化するMA-BERTを提案している[17]。

本研究では、商品属性とレビューの両方を用いて両者の関連付けを行い、言語モデルを構築する。

2.2 テキストからの属性抽出

文章内から属性を抽出する研究がおこなわれている。Iwanariらは、ブログ内の文章から、対象物とその対象物の評価を抽出し、属性に応じて対象物を順位づけた[10]。Fuchsらは、商品説明文と商品の属性名、属性値を与え、商品をあらわすために重要な属性値を抽出した[7]。内田らは、レビューでの相対的な評価や単体での評価から、属性ごとの順序関係を推定し、信頼性の評価を行った[20]。Majumderらは、商品のタイトルから属性を抽出するモデルを双方向LSTMや双方向LSTM-CRFモデルを元にして構築した[11]。Zhengらは、複雑な記述がなされている商品説明文から、知りたい属性の属性値をOpentagモデルを提案し、抽出した[19]。Chenらは、BERTの固有表現抽出を用いて、商品説明文やタイトルから商品属性を抽出した[18]。Wangらは、属性を問いとして与え、属性値を商品説明文から抽出するモデルをBERTをもとに構築した[14]。

本研究では、テキストから属性抽出を行うのではなく、テキストと属性を入力として受け取り、両者の関連を計算するモデルを提案する。

3 問題定義

本節では、本研究で取り組む問題について定義する。ある商品 p の属性集合は (1) のように表すことができる。

$$A = \{(a_1, v_1), (a_2, v_2), \dots, (a_k, v_k)\} \quad (1)$$

ここで k は属性の個数とする。また、 $a_i (1 \leq i \leq k)$ は属性名を表し、 $v_i (1 \leq i \leq k)$ は属性値を表す。カメラを例とすると、属性集合 A は $A = \{(\text{画素数}, 3,410 \text{ 万画素}), (\text{撮像素子}, \text{フルサイズ}), \dots, (\text{重量}, 800\text{g})\}$ のように表すことができる。商品 p に対するレビューを r と置く。レビュー r は、単語の系列である。 i 番目の単語を w_i と置くと、(2) のように表すことができる。

$$r = w_1 w_2 \dots w_L \quad (2)$$

ここで L はレビュー r 中の単語の個数である。本研究では属性集合 A とあるレビュー r を入力として、属性集合 A に対して、レビュー r が書かれる可能性のある文章なのかを予測し、レビュー r と属性集合 A がどの程度関連しているかを関連スコア s で表す。関連スコア s とは、属性集合とレビューがどの程度関連しているか度合いを表す。関連スコア s は、レビュー r と属性集合 A に関連がなければ値が 0 に近くなり、レビュー r と属性集合 A に関連があれば値が 1 に近くなる。この入力と出力の関係を図 1 に示す。

具体的には、 $A = \{(\text{画素数}, 3,410 \text{ 万画素}), (\text{撮像素子}, \text{フルサイズ}), \dots, (\text{重量}, 800\text{g})\}$ のような属性集合が与えられたとする。このカメラは撮像素子がフルサイズであり、画素数が 3,410 万画素であるため、美しく撮影することができると考えられる。このようなカメラの属性集合に対して、「空の青や木々の緑がとてもきれいに撮影できます。」のようなレビューは商品に対する評価としてあり得るテキストであるため、関連スコアは 1 に近くなる。反対に「発色はきれいだが、画質は良いといえない。」

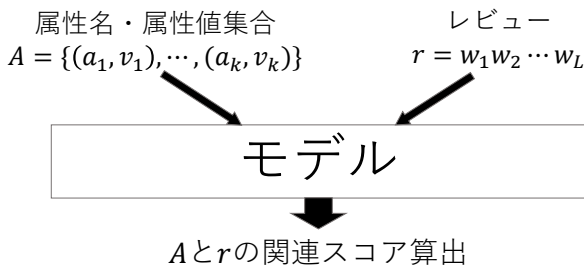


図1 問題定義. レビューと属性名, 属性値集合を入力として, 両者が関連していれば関連スコアは1に近づき, 関連していなければ関連スコアは0に近づく.

のようなレビューは, その商品に対する評価とは考えにくいいため, 関連スコアが0に近くなる. このようなモデルを構築することで, 属性集合 A とレビュー r が対応しているかを明らかにする.

4 提案手法

本節では, 商品の属性とレビューの関連を理解した言語モデルを構築するための提案手法について述べる. 属性値には量的データを含むものがあり, 単純に数値を言語モデルに入力すると値の大小関係や規模感を考慮することができない.

本節ではまず, モデルの概要および本研究で利用する BERT について説明する. その後, 属性値に含まれる量的データを考慮するための仕組みについて説明する. 最後に, モデルの学習方法について述べ, 得られたモデルからどのように関連スコアを計算するか述べる.

4.1 モデルの概要

本節では, 本研究で提案するモデルの概要について述べる. 提案手法の概要について図2に表す. 提案モデルでは入力として, レビュー, 属性集合を受け取り, レビューと属性の関連スコアを算出する. ここで, 関連スコアは入力されたレビューと属性がどの程度関連しているかという度合いを表す. 関連スコアの算出をレビュー, 属性名, 属性値を BERT モデルに入力することで求める. 入力された属性値には, 数値のような量的データを含むものがある. この属性値を適切に入力できるようにするため, 属性値の処理を行い, BERT モデルに入力する. BERT については4.2節で述べる. モデルの学習は, レビューが入力された属性値を持つ商品に実際につけられたレビューであるかを分類する問題として学習させる. 正例として, 入力された属性値を持つ商品に実際につけられたレビュー, 負例として, 異なる属性値を持つ商品に実際につけられたレビューを用いて, この正例と負例を当てるようにモデルを学習させる.

4.2 BERT

本研究では, 様々な自然言語処理タスクで使用される BERT [4] を元に言語モデルの構築を行う. BERT は Transformer En-

coder を用いたニューラルネットワークのモデルである. 大量のテキストでファインチューニングを行うことで様々な自然言語処理タスクを精度よく解けることが知られている. 本研究で提案するモデルを, 図2に示す.

BERT では入力されたテキストを分割し, トークンとして入力する. 本研究では, 既存のトークナイザーで 사용되는トークンに追加して, 属性値を適切に表現するため, 量的データを表すトークンを追加する. 量的データの処理については4.3節で述べる.

また, BERT では, トークンに加え, 文の種類を表す Segment ID が入力される. Segment ID は2つの文章をモデルに入力する際にどの文章が1文目か2文目を区別する目的で使用されることが多い. 本研究では, この Segment ID をレビュー, 属性名, 属性値を判断するために用いる. Segment ID として, レビューには0, 属性名には1, 属性値には2を割り振る. 3種類の Segment ID を用いることで, モデルへの入力テキストに対して, 属性名, 属性値, レビューの判別が可能になる.

レビューと属性を表すテキストの間には, レビューと属性を区別するために境界となるトークンとして [SEP] トークンを加えた.

4.3 属性値に含まれる量的データの処理

本節では, 属性値に含まれる量的データの処理について述べる. 属性値には, 「3,410万画素」や「800g」のように数値データを含む属性が多く存在する. これらの値をモデルに入力してもそれぞれの数値の大小関係や規模感を理解することは難しい [16]. モデルが属性値の量的データを適切に捉えることができるように, 本研究では以下の2つのアプローチを提案する.

- 量的データのカテゴリ化
- 数値の桁数を表すトークンの追加

4.3.1 量的データのカテゴリ化

量的データである数値をカテゴリ化し, テキストとして入力する. 量的データを持つ属性値の大小関係を把握するためにカテゴリ化を行う. 具体的には, 画質が「非常に良いカメラ」や「あまり画質が良くないカメラ」のように, 定めた閾値を元に, 量的データをカテゴリに分類する. カテゴリ化した属性値を表現する方法として, 本研究では2種類の手法を提案する.

1つ目は, 特殊トークンを追加するカテゴリ化の方法である. 画素数や重量など, それぞれの属性において, カテゴリを表すトークンを追加する. 例えば, 画素数を3つのカテゴリに分割するのであれば, 既存のトークンに, [gaso-1], [gaso-2], [gaso-3] のような, 画素数に関するカテゴリを表すトークンを追加する. 属性, 属性値を表すテキストを入力する際には, 「画素数:[gaso-1], 重量:[weight-3] ...」のように, 「属性名:属性値の特殊トークン」の形式で入力する. 量的データを持つ属性に対して, カテゴリ化した数の特殊トークンを追加し, 属性値を表現する. この方法は, 属性ごとにトークンを用意することで, それぞれの属性に合わせた量的データの表現をすることができる.

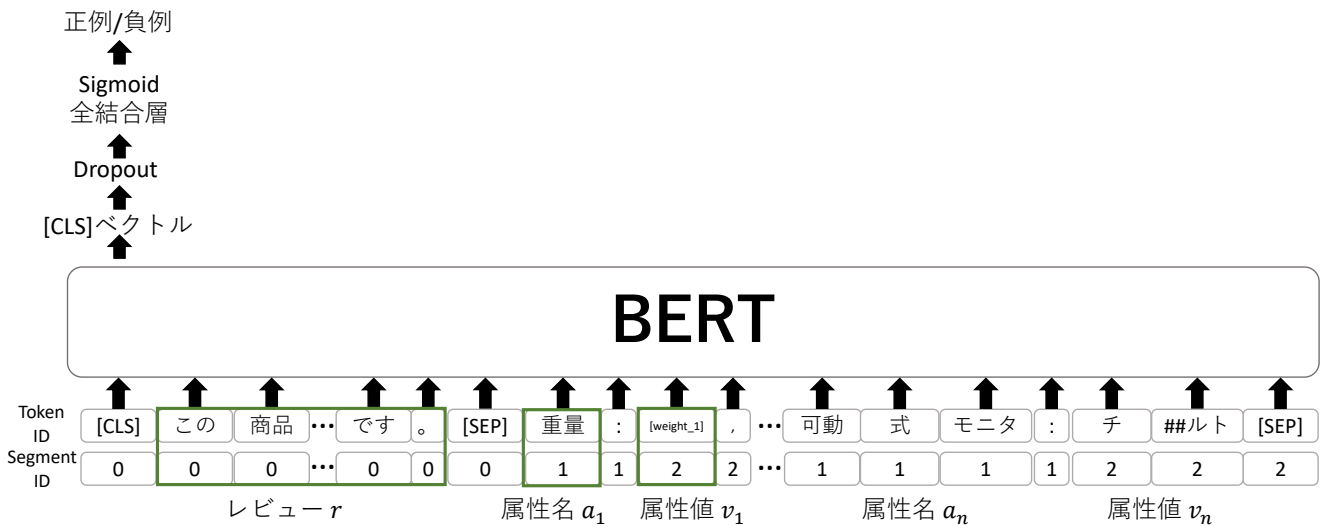


図2 提案モデル. レビューと属性, 属性値を入力として入力した属性値がレビューと対応しているかを判断する提案モデル. 提案モデルは実際に入力した属性値を持つ商品に付けられたレビューを正例, 異なる属性値を持つ商品に付けられたレビューを負例として学習する.

2つ目は, カテゴリを数値を用いて表す方法である. それぞれの属性におけるカテゴリを数値を用いて表現する. トークンを追加する方法では, 特殊トークンが多く追加され, 事前学習済みモデルでは学習されていない単語がトークンが多く出現することになってしまう. そこで, 数値を用いて属性値を表現することで追加トークンを無くし, 属性値の表現をすることができる. 例えば, 画素数を3つのカテゴリに分割するのであれば, 1,2,3の3つの数字を用いる. 画素数が大きいものがカテゴリの値が高くなり, また, 画素数が小さい物は, カテゴリの値が小さくなるようにする. 属性, 属性値を表すテキストを入力する際には, 「画素数:1, 重量:3 ...」のように, 「属性名: 属性値のカテゴリを表現する数値」の形式で入力する.

4.3.2 数値の桁数を表すトークンの追加

属性値がどの程度の大きさかを考慮するために, 数値の桁数を表現する特殊トークンを追加し, 数値の大きさを適切に捉えることができるようにする.

既存の言語モデルでは, 数値をそのまま入力し, 量的データの規模感を理解させることは難しい. Zhangら[16]は, 量的データの規模感を言語モデルで推論するために, NumBERTを提案している. この手法は, 数値の指数表記 [EXP] トークンを追加し, 数値の規模感を表現する手法である. そこで, 本研究では指数表記を表す特殊トークン [EXP] トークンを追加し, 量的データの規模感を表現することで, 量的データを適切に捉えることができると考えた. [EXP] トークンを使用して, どのように数値を表現するのかについて述べる. 例えば重量が3,410万画素のカメラの属性値の数値部分に着目する. このカメラの数値部分は3,410である. 3,410は 3.41×10^3 と表すことができる. この指数表記の10を [EXP] トークンに置き換え, また, 小数点を省略する. すると, 3410という数字は341[EXP]3と

いうように表現することができる. このように属性値の量的データの規模感を捉えるために, 特殊トークン [EXP] トークンを用いる.

属性, 属性値を表すテキストを入力する際には, 「画素数:341[EXP]3, 重量:8[EXP]2 ...」のように, 「属性名:[EXP] トークンを用いて表現した属性値」の形式で入力する.

4.3.1節, 4.3.2節で述べた属性と属性値の入力形式について, 表1に表す.

4.4 正例と負例の構築

本節では, 正例と負例をどのように構築するかを述べる. モデルは, レビューに対して, 入力された属性値が関連しているかを当てるように学習を行う. この学習を行う上で, 正例として入力された属性値を持つ商品に実際につけられたレビュー, 負例として異なる属性値を持つ商品に付けられたレビューを当てられるように学習する. このような学習を行うことで, レビューと商品の属性集合の対応を理解することができる.

負例として用いる, 異なる属性値を持つ商品の選び方について述べる. 異なる商品名を持つすべての商品を負例として用いると, 後継商品やキットとして付いているレンズの違いなどあまり属性値の違いがない商品を負例として用いる可能性がある. 属性値が極端に類似している商品を負例として学習に用いると, 属性値とテキストが似ているのにも関わらず負例となり, モデルが正しく属性値やレビューの情報を捉えられないため, 学習することが難しいと考えられる. そこで, 本研究では商品ごとにコサイン類似度を求め, そのコサイン類似度が一定の値以下の商品を負例として用いる.

コサイン類似度の求め方について説明する. 属性値については, カテゴリカルデータと量的データの2種類がある. カテゴリ

表 1 量的データを含む属性値の表現方法.

属性値の表現方法	入力形式
量的データをそのまま入力	画素数:3410, 重量:800 …
特殊トークンを用いたカテゴリ化	画素数:[gaso-1], 重量:[weight-3] …
数値を用いたカテゴリ化	画素数:1, 重量:3 …
数値の桁数を表す特殊トークンの使用	画素数:341[EXP]3, 重量:8[EXP]2 …

リカルデータについては、ダミー変数に変換する。量的データについては、標準化を行う。このように属性値を変換し、コサイン類似度を求め、レビューに対して負例として用いる商品を選択する。

4.5 モデルの学習

本研究の提案手法では、入力された商品の属性値に対して、入力されたレビューがその属性値を持つ商品に付けられる可能性のあるレビューかを予測する。

学習を行ううえで、正解との誤差を計算し、損失値を求め、逆伝播を行う。損失関数として、バイナリ交差エントロピーを用いた。バイナリ交差エントロピーの式を表す。いま、 o_i を訓練データ内の i 番目の商品に対して、正例であるとモデルが予測した確率とする。また、 y_i を i 番目の商品の正解の値とする。商品の正解の値とは、入力された属性値を持つ商品に付けられたレビューであるとき y_i は 1、そうでなければ y_i は 0 とする。

訓練データの個数を N 、 i 番目のデータに対して正例であると予測した確率を o_i 、 i 番目のデータの正解の値を y_i とするとき、以下のバイナリ交差エントロピーを最小化することでモデルを学習する。

$$L = - \sum_{i=1}^N (y_i \log o_i + (1 - y_i) \log(1 - o_i)) \quad (3)$$

4.6 関連スコアの計算

提案モデルでは、入力された商品属性とレビューが対応しているかを関連スコアと定義し、この関連スコアを求める。具体的には、モデルに属性と属性値集合 A とレビュー r を入力することで、両者が対応している確率を得ることができる。この確率を、属性値とレビューが対応している度合い、関連スコアとして用いる。

関連スコアは、シグモイド関数を用いることで求められる。ある商品の属性集合 A とレビュー r を用いて、関連スコア s を求める。この関連スコア s はシグモイド関数を用いることで、 $(0, 1)$ の間の値となるため、確率として表現できる。

5 実 験

本節では、提案手法の有効性を検証するために行う実験について述べる。

5.1 データセット

本研究では、デジタル一眼レフカメラを対象にし、価格.com¹

表 2 使用するカメラの属性.

属性名	形式	属性値の例
メーカー	文字列	CANON, SONY
タイプ	文字列	ミラーレス, 一眼レフ
画素数	数値	1600 万
撮像素子	文字列	フルサイズ, APS-C
最低撮影感度	数値	200
最高撮影感度	数値	25600
最低シャッタースピード	数値	1/16000
最高シャッタースピード	数値	60
撮影枚数	数値	210
防塵・防滴	文字列	あり, なし
幅	数値	106.5
高さ	数値	64.6
奥行	数値	33.3
可動式モニタ	文字列	チルト, バリアングル
重量	数値	240
カラー	文字列	ブラックシルバー

のレビューと商品スペック表を用いて実験を行う。デジタル一眼レフカメラは、様々な属性を持ち、スペック表を見ただけではどのような商品かを理解することが難しい。またレビューからの属性に言及されているものなのかを判断することは難しい。レビューと商品の属性の対応関係を理解することが難しいため、本研究ではデジタル一眼レフカメラを対象とする。

価格.com のページでは、デジタル一眼レフカメラとして、一眼レフ、ミラーレス一眼が含まれている。対象商品数は 508 点、また、総レビュー数は 23,381 件である。

本節では、使用したデータの詳細と属性やレビューに対する前処理、正例と負例の定義について述べる。

5.1.1 商品属性

本研究では 17 種類の属性を対象として、実験を行った。これらの属性を選択した理由は、これらの属性に関するレビューが多く、ユーザの多くがこの属性に着目して商品を選ぶと考えたからである。

属性値において数値を含むものとして、画素数、最低撮影感度、最高撮影感度、最低シャッタースピード、最高シャッタースピード、撮影枚数、幅、高さ、奥行、重量が挙げられる。これらの属性に対して、数値のカテゴリ化を行った。1 カテゴリにおける商品数が同数になるようにカテゴリ化を行った。4.3.1 節で述べたカテゴリ化のため、画素数、最低撮影感度、最高撮影感度、最低撮影枚数、最高撮影枚数、撮影枚数、幅、高さ、奥行、重量は、属性値が欠損値のものを含む 5 種類のカテゴリに分類した。最低シャッタースピード、最高シャッタースピード

1: <https://kakaku.com/>

は欠損値のものを含む 8 種類のカテゴリに分類した。

5.1.2 レビュー

レビュー中のテキストの中には、「後継機として買いました。」や「セールだったので購入しました。」のような購買情報に関するテキストもある。これは商品の属性に関するテキストではないので、学習として用いるのはふさわしくない。本研究では、どのようなテキストが商品の属性を表現するために用いられるテキストなのかを学習することを目的としているため、このようなテキストを学習するのはふさわしくない。

そこで、商品の属性に言及したレビューを抽出するために、評価項目ごとに書かれたレビューを使用する。価格.com では、評価項目ごとにレビューが書かれている。例えばカメラであれば、「デザイン」、「画質」、「操作性」、「バッテリー」、「携帯性」、「機能性」、「液晶」、「ホールド感」の 8 つの評価項目が挙げられる。画質であれば、撮影感度や画素数、撮像素子などに関連しており、また、携帯性であれば、大きさや重さなどの属性が関連していることが考えられる。このように評価項目に基づくレビューには、属性に関連したレビューが書かれていることが考えられる。そこで、本研究ではこの評価項目に基づいたレビューを抽出する。

価格.com では、この評価項目に基づくレビューを、評価項目を「【】」で囲み、その後が続いてレビューを書いている。具体的には、「【画質】 3,300 万画素でもかなり高精細。」のように、「【】」の中に評価項目を記述し、そのテキストに続いて、商品のレビューが書かれている。このような形式で価格.com の多くのレビューは書かれている。本研究では、学習に用いるレビューとして複数文に分かれているレビューについては、1 文ずつ抽出する。

この抽出したレビューの長さは、長い文章から短い文章まで多様な長さの文章が抽出できる。短すぎる、あるいは長すぎるテキストを除去するため、本研究では、全文書の中で 25% から 75% の長さに含まれるレビューを学習に用いる。25% から 75% となるレビューの文字数は 19 文字以上 47 文字以下であった。この条件でレビューを取得すると 107,917 文取得できた。

5.1.3 正例と負例の定義

本節では、正例と負例の定義について述べる。今回はある属性集合を持つカメラに対して、実際についたレビューと、そのカメラの属性集合を正例とする。また負例は、異なる属性集合を持つカメラに対して、実際についたレビューと、異なる属性集合とする。負例となる商品は、コサイン類似度が 0.95 以下の商品をランダムで選択する。コサイン類似度が 0.95 以下の商品からランダムで商品を選択し、似ている属性値を持たない商品を負例とする。

5.2 実験設定

BERT の事前学習済みモデルは東北大学乾研究室の事前学習済みモデルを使用する²。学習率は 2.0×10^{-6} とした。訓練データを学習するエポック数については、検証データに対して

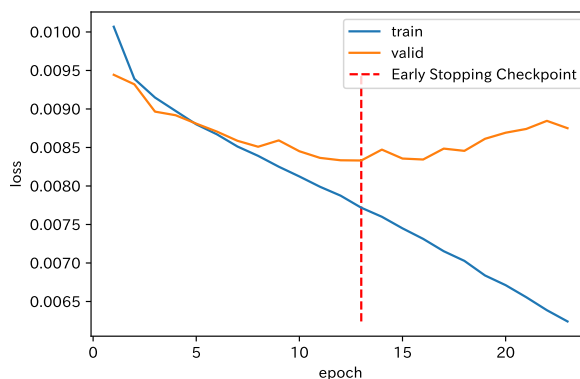


図 3 量的データを数値を用いたカテゴリ化し Segment ID を 3 種類にして入力した時の学習中の損失値の変化。赤点線は早期終了したエポック数を表す。

Patience を 10 に設定した早期終了に基づき決定した。

レビューと属性集合を入力したトークン数は、256 トークン以下に設定した。訓練データ、検証データは 425 商品のレビューを対象とした。訓練データ 87,165 件、検証データ 9,685 件として学習を行った。また、評価用のデータは 50 商品 11,067 件のテキストを用いる。訓練データ、検証データには評価用データと同じ属性値を持つ商品は含まれていない。

量的データを数値を用いてカテゴリ化し、Segment ID を 3 種類にして学習した際の学習中の損失値の変化を図 3 に示す。

5.3 比較手法

本節では、今回比較を行う手法について述べる。本研究では以下の手法の精度を比較した。

- 特殊トークンを用いた属性値のカテゴリ化

4.3.1 節で述べた量的データを含む属性値を、特殊トークンを用いてカテゴリ化を行う手法。

- 数値を用いた属性値のカテゴリ化

4.3.1 節で述べた量的データを含む属性値を、数値を用いてカテゴリ化を行う。

- 数値の桁数を表す特殊トークンの使用

4.3.2 節で述べた数値の規模感を理解するために、指数表記を表す特殊トークン [EXP] トークンを追加し、属性値を表現する方法。

- 属性値を数値のまま入力

ベースライン手法として、属性値の数値をカテゴリ化などをせずにそのまま入力するモデルを用いた。例えば、「画素数:3410, 重量:800 …」のように、言語モデルの入力に量的データをそのまま入力する。

それぞれの属性値の入力方法について、Segment ID をレビューを 0、属性、属性値を 1 とおく、2 種類にする方法とレビューを 0、属性を 1、属性値を 2 とおく 3 種類で表現する方法を比較する。

本研究で使用する、東北大学乾研究室の事前学習済みモデルの BERT は、Segment ID が 2 種類、0 と 1 のみを持つ時の重

2 : <https://github.com/cl-tohoku/bert-japanese>

表 3 提案手法ごとの正解率の比較.

Segment ID の数	属性値の表現方法	正解率
2 種類	量的データを数値のまま入力	0.687
3 種類	量的データを数値のまま入力	0.677
2 種類	特殊トークンを用いたカテゴリ化	0.693
3 種類	特殊トークンを用いたカテゴリ化	0.697
2 種類	数値を用いたカテゴリ化	0.681
3 種類	数値を用いたカテゴリ化	0.714
2 種類	数値の桁数を表すトークンの使用	0.703
3 種類	数値の桁数を表すトークンの使用	0.687

表 4 Segment ID が 3 種類, 属性値の量的データを数値を用いたカテゴリ化を行ったモデルの混同行列.

		予測ラベル	
		関連あり	関連なし
正解ラベル	関連あり	3,731	1,803
	関連なし	1,362	4,171

みが学習されている. 本研究では, 属性値の Segment ID として 2 を用いるため, Segment ID が 0 であるときと 1 であるときの重みを平均し, その重みを Segment ID が 2 であるときの重みの初期値として, ファインチューニングを行った.

5.4 評価方法

本節では評価方法について述べる. 本研究では入力された属性値が実際に入力された属性値を持つ商品についてのレビューかを判断できるかを求め, 正解率を用いてモデルの精度を比較する.

5.5 実験結果

本節では, 実験結果について述べる. 学習したそれぞれのモデルに対して, 入力された属性値とレビューが関連しているものなのかを分類するタスクを与え, その正解率を確認する. 5.2 節で述べた 50 商品 11,067 件のテキストを用いて評価を行う. それぞれの手法に対する正解率を表 3 に示す.

ベースラインである, 量的データを数値のまま入力する手法と比較し, 提案手法では正解率が上回ることが分かった. その中でも, Segment ID をレビュー, 属性名, 属性値にそれぞれ割り振り, 数値を用いてカテゴリ化を行う手法が最も正解率が高く, 正解率は 0.714 であった. この手法での予測結果の混同行列を表 4 に示す.

混同行列から, 入力された属性値と関連していないものを関連していないとよく分類できていることがわかる. 量的データを数値を用いたカテゴリ化を行った際には, 高い精度で対応関係を当てることができることがわかる.

5.6 実例の分析

本節では, 最も正解率の高かった量的データを数値を用いたカテゴリ化した手法において, どのような属性とテキストが対応関係を正しく予測できているのか, また対応関係を正しく予測できていない属性とテキストはどのようなものがあるのか

分析していく. まず, はじめに対応関係が関連ありであり, その予測を正しくできたものについて見ていく. 商品の画素数が 4575 万画素, 撮像素子がフルサイズの商品に対して「初代とほぼ変わりなく非常に安定した高画質といえます。」というテキストを与えた. これは実際にこの商品についてのレビューであり, それを正しく関連ありと予測することができた. 高画質という単語が出てくるテキストを画素数が高いカメラに対して, 関連があると予測できている.

次に, 対応関係がなく, それを正しく関連がないと予測できているものについて見ていく. 属性として撮像素子がフォーサーズである商品を入力する. テキストは「他 APS-C 機と比較しても悪いということはありません。」を入力する. 「APS-C」という異なる撮像素子がテキストに入力されている場合, それを考慮して関連がないと判断できることが分かった.

最後に正しくテキストと属性の関連を当てられなかったものについて分析していく. 画素数が 2410 万画素で撮像素子が APS-C 機のカメラの属性を入力とする. それに対して, 実際についてのレビューである「最近の APS - C 機種は 2400 万画素で画質は良好ですが, 高感度対応力はそれなりです。」というテキストを与える. これは, 実際についてのレビューなので対応関係があるが, これを関連なしと予測した. 属性を入力する際に数値でカテゴリ化をしたため, テキスト内の「2400 万画素」との大きさの関係を理解することが難しかったと考えられる.

5.7 考察

本研究では, 商品の属性名, 属性値とレビューの対応関係を学習させることによってそれぞれの関連を理解した言語モデルを構築した. 入力形式を変更することにより, 対応関係を当て精度の違いを比較した. 数値を用いたカテゴリ化は Segment ID を 2 種類とした時と比較して, Segment ID を 3 種類にした時の正解率は高くなった. これは, Segment ID が 3 種類の時に, 数値となる箇所が属性値だと判断できるため精度が上がったと考えられる. また, 特殊トークンを用いたカテゴリ化を行った際には, Segment ID が 2 種類であるときも 3 種類であるときも精度があまり変わらなかった. これは, 特殊トークン自体が属性値の情報を持つため, 精度が変わらなかったことが挙げられる. モデルの精度の比較をし, ベースラインである Segment ID が 2 種類で量的データを数値のまま入力する方法よりも量的データの入力を工夫したほうが精度が高くなることが分かった.

今回は評価として, 学習したモデルが対応関係を理解しているかによって精度を比較した. 学習を行った言語モデルが属性集合に対してレビューとの評価ができているか, 関連を捉えることができるのかを求める必要がある. 今後はモデルが属性値とテキストの関連を求めることができるのか評価する必要がある. 対応関係だけでなく, 言語モデルの評価を今後行う.

また, 本研究では属性値の入力形式を工夫することで商品属性を適切に捉える言語モデルの構築に取り組んだ. モデルの構造として, 属性に情報を捉えることができれば, 精度が高くなると考えられる. より属性の情報を考慮するモデルの構造を検

討する必要がある。

6 ま と め

本研究では、テキストである経験的属性と商品の属性名、属性値といった探索的属性の対応関係を理解した言語モデルを構築する。このモデルにより、属性名や属性値と関連している評価なのかを判断することができ、探索的属性である商品の属性名、属性値と経験的属性であるテキストの対応関係を理解することができる。

事前学習済みの BERT モデルを用いて、量的データを含む商品の属性値を適切に捉えるため、カテゴリ化や規模感を表す特殊トークンの追加といった入力形式の工夫を行い、複数の手法を提案した。

量的データを数値のまま入力するベースラインと比較して、量的データに入力の工夫を行った提案手法のほうが精度が高いことが分かった。また、対応関係を理解しているか評価を行い、量的データを数値としてカテゴリ化する手法が最も正解率が高かった。今回は対応関係の学習を行い、対応を正しく当てられるかを評価した。今後はレビューと属性を入力として、関連しているかどうかスコアを算出し、実際に属性とテキストの関連を捉えられているのか評価していく。

謝辞 本研究は JSPS 科学研究費助成事業 JP21H03775, JP22H03905, ならびに 2022 年度国立情報学研究所共同研究 22S1002 による助成を受けたものです。ここに記して謝意を表します。

文 献

- [1] Qibin Chen, Junyang Lin, Yichang Zhang, Hongxia Yang, Jingren Zhou, and Jie Tang. Towards knowledge-based personalized product description generation in e-commerce. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 3040–3050, 2019.
- [2] Shiqian Chen, Chenliang Li, Feng Ji, Wei Zhou, and Haiqing Chen. Review-driven answer generation for product-related questions in e-commerce. *arXiv preprint arXiv:11905.01994*, 2019.
- [3] Yang Deng, Wenxuan Zhang, and Wai Lam. Opinion-aware answer generation for review-driven question answering in e-commerce. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pp. 255–264, 2020.
- [4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- [5] Li Dong, Shaohan Huang, Furu Wei, Mirella Lapata, Ming Zhou, and Ke Xu. Learning to generate product reviews from attributes. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pp. 623–632, 2017.
- [6] Yue Feng, Zhaochun Ren, Weijie Zhao, Mingming Sun, and Ping Li. Multi-type textual reasoning for product-aware answer generation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1135–1145, 2021.
- [7] Gilad Fuchs, Haggai Roitman, and Matan Mandelbrod. Automatic form filling with form-BERT. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1850–1854, 2021.
- [8] Shen Gao, Xiuying Chen, Zhaochun Ren, Dongyan Zhao, and Rui Yan. Meaningful answer generation of e-commerce question-answering. *ACM Transactions on Information Systems*, Vol. 39, No. 2, pp. 1–26, 2021.
- [9] Shen Gao, Zhaochun Ren, Yihong Zhao, Dongyan Zhao, Dawei Yin, and Rui Yan. Product-aware answer generation in e-commerce question-answering. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pp. 429–437, 2019.
- [10] Tatsuya Iwanari, Naoki Yoshinaga, Nobuhiro Kaj, Toshiharu Nishina, Masashi Toyoda, and Masaru Kitsuregawa. Ordering concepts based on common attribute intensity. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI'16*, p. 3747–3753. AAAI Press, 2016.
- [11] Bodhisattwa Prasad Majumder, Aditya Subramanian, Abhinandan Krishnan, Shreyansh Gandhi, and Ajinkya More. Deep recurrent neural networks for product attribute extraction in ecommerce. *arXiv preprint arXiv:1803.11284*, 2018.
- [12] Julian McAuley and Alex Yang. Addressing complex and subjective product-related queries with customer reviews. In *Proceedings of the 25th International Conference on World Wide Web*, pp. 625–635, 2016.
- [13] Kalyani Roy, Smit Shah, Nithish Pai, Jaidam Ramtej, Prajit Nadkarni, Jyotirmoy Banerjee, Pawan Goyal, and Surenender Kumar. Using large pretrained language models for answering user queries from product specifications. In *Proceedings of The 3rd Workshop on e-Commerce and NLP*, pp. 35–39, 2020.
- [14] Qifan Wang, Li Yang, Bhargav Kanagal, Sumit Sanghai, D Sivakumar, Bin Shu, Zac Yu, and Jon Elsas. Learning to extract attribute value from product via question answering: A multi-task approach. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 47–55, 2020.
- [15] Shiwei Zhang, Jey Han Lau, Xiuzhen Zhang, Jeffrey Chan, and Cecile Paris. Discovering relevant reviews for answering product-related queries. In *2019 IEEE International Conference on Data Mining*, pp. 1468–1473, 2019.
- [16] Xikun Zhang, Deepak Ramachandran, Ian Tenney, Yanai Elazar, and Dan Roth. Do language embeddings capture scales? In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4889–4896, 2020.
- [17] You Zhang, Jin Wang, Liang-Chih Yu, and Xuejie Zhang. MA-BERT: Learning representation by incorporating multi-attribute knowledge in transformers. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 2338–2343, 2021.
- [18] Chen Zhao, Yuki Nakayama, and Koji Murakami. Multi-task learning for product information with fine-tuned BERT. 言語処理学会 第 26 回年次大会 発表論文集, pp. 736–739, 2020.
- [19] Guineng Zheng, Subhabrata Mukherjee, Xin Luna Dong, and Feifei Li. Opentag: Open attribute value extraction from product profiles. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 1049–1058, 2018.
- [20] 内田臣了, 山本岳洋, 加藤誠, 大島裕明, 田中克己. 不確かな順序関係の分析に基づく経験的属性によるオブジェクト検索. 第 8 回データ工学と情報マネジメントに関するフォーラム, C2-4, 2016.