# 分散表現を用いた対話型 不満調査データセット検索システム

原田 裕生　　末廣 駿　　斎藤 博昭
慶應義塾大学　理工学部情報工学科
{harada, suehiro, hxs}@nak.ics.keio.ac.jp

## 1 はじめに

企業にとって利用者が製品やサービスに対してどのような反応を示しているかはその後の製品開発や経営戦略を考えていく上での重要な情報資源である。とりわけサービス製品に対する利用者の不満は、企業側がサービス改善を図る際の具体的な助けとなる。また、既存のサービスや製品に対して、利用者がどのような不満を抱いているのかを調査すれば新しい製品やビジネスチャンスにつながる可能性もある。

こういった背景の中、利用者から不満を買い取り、それを必要とする企業に売るというビジネスが株式会社不満買取センターによって行われている。この不満買取センターに一般ユーザーが投稿した様々な不満は、研究者向けに不満調査データセット[1]として提供されている。このデータセットは不満買取センターのオペレーターによってタグ付けがなされているが、膨大な生データであり、有効なデータを抽出するには言語処理をする必要がある。

そこで本稿では、検索をする際に、ユーザーが検索したいキーワードに対して、キーワードと一致するデータのみを抽出する仕様ではなく、キーワードに似ている単語や、関連度の高い単語を含むデータに関しても抽出を行う検索システムを設計することを目的とする。

さらに、類義語・関連語に関しては検索に含むかどうかの判断を対話型で取り入れることにより、検索者のイメージするキーワードでの検索が可能になる設計とした。また、実験を通して精度の確認を行う。また、本稿とは別に「生データを様々な言語処理に対応しやすくすること」を目的にした論文[1]があり、検索システムがあるため精度の比較をする。

## 2 不満調査データセット

本稿と従来システムで扱う不満調査データセットには254,683件の不満が含まれており、json形式で提供されている。不満が投稿された機関は2015年3月18日から2015年9月23日までである。1件の不満は表1のようにタグ付けされた複数の情報を持つ。

表1：各不満が持っているタグの一部

| タグ名 | 説明 |
|---|---|
| fuman(必須) | ユーザーが投稿した不満そのもの |
| category | メインカテゴリ |
| sub_category | サブカテゴリ |
| company_name | 不満の対象となっている企業 |
| product_name | 不満の対象 |

## 3 関連研究

従来システム[1]は、素性化の際に素性としてcategoryとsub_category、company_name、product_name、fumanのタグを用いた。各タグについて形態素解析JUMAN[2]および構文解析ツールのKNP[3]を用いて解析を行った後、正規化代表表記、カテゴリ・ドメイン、Wikipedia中のエントリ・リダイレクト・上位語の意味情報を素性として取り出す。ただし、fumanに関しては、構文解析の結果から主辞形態素の品詞が

- 動詞または形容詞
- 数詞や代名詞でない名詞

である単語についてのみ意味情報を取り出す対象とした。

各タグからから取り出した意味情報をまとめ、重複を省いたものを一つの不満の素性とし、それと単語辞書を照らし合わせて素性ベクトルを作った。単語辞書は全ての不満から得た素性（単語）をまとめたものである。どのタグから得た素性なのかにはこだわらず重複を除いたものを単語辞書として使用した。

[1]本稿では、株式会社不満買取センターが国立情報学研究所の協力により研究目的で提供している「不満調査データセット」を利用させていただいた。

## 4 システム構成

本節では扱うツールについて説明し、提案方法について述べる。

### 4.1 word2vec

本稿では特定の単語に対する類義語・関連語を抽出する際に word2vec を用いた。word2vec[4]とは Mikolov らによって提案された、ニューラルネットワークを用いて単語の分散表現を獲得する手法である。正確には、その手法が実装されたオープンソースソフトウェアの名称である。この手法では学習を実現するためのネットワーク構造として、Continuous Bag-of-Words(CBOW) モデルと skip-gram モデルの二つのモデルを提案しているが、本稿では CBOW モデルでの学習を行った。CBOW モデルはある単語 $w_t$ に対して周辺単語 $w_{t-k}$, …, $w_{t-1}$, $w_{t+1}$, …, $w_k$ を入力とし、$w_t$ を出力とする(予測する)ニューラルネットワークである。入力と出力は 1-of-K 単語ベクトルとなっている。中間層は隠れ変数となっており、任意の個数のノードを設定することができる。CBOW モデルでは計算の簡略化のために単語 $w_t$ の前後に出てくる単語の語順を無視しているため、文法的な意味を加味することはできない。word2vec ではテキスト集合である学習データの文脈をもとに単語の関係性をベクトル化して扱うことを可能にする。

word2vec のパラメーターとしては

学習率　0.01
周辺単語の数 5
学習モデル　CBOW
分散表現の次元数 200
最低出現回数　5

とした。

### 4.2 JUMAN

word2vec で日本語単語ベクトルのモデルを作成する際に、文を単語ごとに分かち書きする必要があるため、本稿では形態素解析ソフト JUMAN を用いた。さらには JUMAN の特性から

- 代表表記
  基本語彙の表記ゆれを吸収する
  ex.）子供/子ども/こども　→ 子供
- 品詞による選別
  ・動詞または形容詞
  ・数詞や代名詞でない名詞
  のみを抽出

を行うことによって、同じ意味の単語が表記揺れによって別単語として認識されることを防ぎ、単語ベクトルを構築する際に不利な要因を減らすことができる。

### 4.3 日本語単語ベクトルの構築

#### 4.3.1 不満由来の単語モデル

不満データセットで、1 件 1 件の不満において 2 節で挙げた全タグについて形態素解析を行い、一つの文章として扱い単語ベクトルを構築した。全タグをまとめて一つの文章として扱うことで、企業名やカテゴリ、不満本文内の出現単語の意味上の繋がりを生み出すことを意識した。

#### 4.3.2 Wikipedia 由来の単語モデル

Wikipedia のダンプデータ(01/03/2018)を用いて不満由来の単語ベクトルと同様の条件で構築した。

## 5 実験

### 5.1 検索のプロセス

以下に検索の流れを示す。

1. 入力から単語を抽出して検索リストを作成する
2. 1 の検索リストに含まれる単語に対して Wikipedia 由来の単語ベクトルを用いて類義語を出力し、類義語を検索リストに加えるか否かをユーザーが選択
3. 1 の検索リストに含まれる単語に対して不満由来の単語ベクトルを用いて類義語を出力し、類義語を検索リストに加えるか否かをユーザーが選択
4. 検索リストに含まれるキーワードを文字列として含む不満データを検索結果候補とする
5. 検索結果候補に関して、各不満データ内の「キーワードの出現回数」と「不満本文の長さ」を尺度としスコアづけをし、スコアが高い上位のものを検索結果として出力

入力は文もしくは単語、また複数の文を検索する場合は半角スペースを区切りとして入力することを想定している。入力に用いられる文字は日本語か英数字で、記号類は対象外とした。

## 5.2 類義語追加例

以下に入力からの単語抽出と対話型の、検索リストへの類義語の追加の例を示す。

入力 ： 政治
抽出した単語 ：[ '政治' ]
(Wikipedia による類義語の追加)
経済を追加しますか？
Yes
外交を追加しますか？
No
ハーシムを追加しますか？
No
財政を追加しますか？
No
検索リスト[ '政治', '経済' ]
(不満による類義語)
独立を追加しますか？
No
国を追加しますか？
Yes
政府を追加しますか？
Yes
安保を追加しますか？
No
自衛を追加しますか？
No
検索リスト[ '政治', '経済', '国', '政府' ]

## 5.3 実験設定

本来のシステムでは約 25 万件全てのデータに関して検索が可能だが、従来システムとの比較と、検索の精度の評価を優先したため

- 25 万件の中から無作為に抽出した 200 件に対して検索を行った
  （従来システムでは 2.5 万件から無作為の 200 件）
- 入力を「衣類」「スポーツ」「金銭」「乗り物」「時計」「値段が高い」とした（従来システムと同様）
- 出力結果に関して適合率・再現率・f 値を尺度として評価をした。

## 5.4 結果

本システムと従来システムの結果を以下の表に示す。

表 2： 本システムの実験結果

| 入力 | 適合率 | 再現率 | f 値 |
|---|---|---|---|
| 衣類 | 1.00 | 0.63 | 0.77 |
| スポーツ | 0.75 | 1.00 | 0.86 |
| 金銭 | 0.72 | 0.87 | 0.79 |
| 乗り物 | 0.64 | 0.70 | 0.67 |
| 時計 | 1.00 | 1.00 | 1.00 |
| 値段が高い | 0.90 | 0.90 | 0.90 |

表 3： 従来システムの実験結果[1]

| 入力 | 適合率 | 再現率 | f 値 |
|---|---|---|---|
| 衣類 | 0.84 | 0.61 | 0.71 |
| スポーツ | 0.48 | 1.00 | 0.65 |
| 金銭 | 0.93 | 0.70 | 0.80 |
| 乗り物 | 1.00 | 0.67 | 0.80 |
| 時計 | 1.00 | 0.93 | 0.97 |
| 値段が高い | 1.00 | 1.00 | 1.00 |

## 5.5 評価

入力が「衣類」「スポーツ」「時計」に関しての結果は本稿のシステムの方が概ね高い精度を示したのに対し、「金銭」「乗り物」「値段が高い」に関しては従来ステムの結果を下回る結果となってしまった。
誤出力となったデータは

- 「〜がくさい」の「がく」から「額」(≒値段)
- 「スピーカー」の「カー」から「自動車」(≒乗り物)

といった文字列の一致を検出したためであると考えられる。
必要な類義語を検出できず検索失敗したものは、類義語を追加するプロセスで、検索リストに含まれているそれぞれのキーワードに関して類似度が高い「2 件」を検索リスト追加リストとしてユーザーに問いたため、類似度が低い類義語は検索リストの追加タスクから漏れていたと考えられる。
精度が上がったものに対しては、類義語の検索リストへの追加の際、対話型を設けたことで、ユーザーの意図に柔軟に対応した検索リストの作成ができたためだと思われる。

## 6 考察

5 節の、類義語抽出の精度は高く、類義語追加のプロセス・検索のプロセスにより誤出力や検索失敗が発生したと考えられる。こういったことから検索の精度を上げるためには

- 検索リストに含まれるキーワードと類似度が高い類義語の数を現システムよりも多くすることで類義語の追加漏れを減らす
- 検索の際に、検索リストに含まれる単語と不満データセットを「文字列の一致」として比較するのではなく、不満データセットを形態素解析し、リストとの「単語の一致」として比較することで不本意な文字列の一致をなくす

ことで誤出力・検索失敗を減らすことができると考える。

## 7 まとめ

本稿では不満調査データセットの検索において、類義語に関しても検索できる手法の提案とシステムの構築を行った。結果、類義語の抽出に関しては高い精度での開発ができた。また、従来システムと比較して遜色ないシステムになった。今後は考察で述べた方法を試すほか、データの前処理なども模索し、より頑強なシステムを構築していく予定である。

## 参考文献

[1] 末廣駿, 斎藤博昭“不満調査データセットの素性ベクトル化”, 言語処理学会第 23 回年次大会発表論文集, p545-p548, 2017.

[2] 黒橋禎夫, 河原大輔, “日本語形態素解析システム JUMAN version5.1”, 2005.

[3] Kawahara, D. and Kurohashi, S., “A fully lexicalized probabilistic model for Japanese syntactic and case structure analysis”, in Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, pp. 176–183, Association for Computational Linguistics, 2006.

[4] Tomas Miklov, Kai Chen, Greg Corrado, and Jefferey Dean. Efficient estimation of word representations in vector space. In Proceedings of Workshop at ICLR, 2013.