

# 文の極性情報を考慮した単語分散表現による ニューラル極性分類モデルの事前学習

田代 光                      鈴木 良弥

山梨大学大学院 医工農学総合教育部 メカトロニクス工学コース  
{g17tj009, ysuzuki}@yamanashi.ac.jp

## 1 はじめに

Webの普及に伴いSNSやレビューサイトなどのユーザ個人が気軽に情報を発信できる場が増加している。ユーザによって提供された情報は他のユーザにとって有益な情報を含むことが多く、その様なテキストを評価極性ごとに分類を行うことは商品推薦システムなどにおいて提供された情報を分析する際に重要となる。

テキスト分類の手法としては文章や文に付与された分類ラベルを用いた教師あり学習手法が一般的であり、近年ではディープニューラルネットワークを用いた分類モデルが優れた結果を収めている [9][10][11]。これらのモデルでは単語の分散表現あるいは埋込み表現と呼ばれる密な実数値ベクトルを入力とすることで学習を行う。各単語はモデルの埋込み層においてベクトル表現へと変換されてモデル入力となる。この埋込み層における各単語のベクトル表現はランダムな値あるいは、Word2Vec[7]に代表されるような学習モデルを用いて大規模コーパスから事前学習したものを利用して初期化される。Iyyerら [4]は評価極性分類において事前学習された単語分散表現は極性を持つ語(感情語)の特徴を上手く捉えられるために、ランダムな値で初期化する場合と比べて分類精度が向上すると考察している。

本研究では評価極性分類の精度向上を目指し、文の極性情報を考慮した単語分散表現 [13][14]を初期化に用いることでその有用性を検証する。具体的には、異なる手法で事前学習された単語分散表現を分類モデルの埋込み層の初期値として用いてレビュー文の評価極性分類を行う。

## 2 単語分散表現

単語の意味表現の最も単純な表現は各単語を出現語彙と等しい長さのベクトルで表現して対応する単語の

要素のみを1その他の要素を0とするone-hot表現である。しかし、one-hot表現では出現語彙の索引のみを符号化するため、語彙間の関係構造を捉えることができない。この問題を解決するために分散表現と呼ばれる低次元で密な実数値ベクトルで表現する研究が数多く行われてきた [1][7][8]。

一般的な分散表現の学習手法は、単語の表現が文脈に反映されるという分布仮説 [2]に基づいて、テキストに出現する単語とその局所的な文脈語を利用して学習を行う。文法的あるいは意味的に類似する単語を隣接したベクトル空間に埋込むことで単語間の意味的な類似性・関連性を捉えた分散表現を獲得できる。一方で、評価極性分類において重要な情報となる感情語のような肯定/否定の極性を持つ語は、商品レビューなどの評判情報を含んだテキストにおいては類似した文脈に出現する。そのため、文脈情報のみから学習された単語の分散表現では『良い』『悪い』のような異なる極性を持った語が隣接して埋込まれてしまう。

Labutovら [6]らは極性ラベルの付与されたレビューデータを用いて単語の文脈情報から学習した分散表現を調整することで極性分類に適した分散表現の学習を行っている。また、Tang[12][13]らはCollobertら [1]の学習モデルを基に単語と文脈語の関連性を推定すると同時に文の極性を推定することで肯定/否定の極性を考慮した分散表現の学習モデルを提案している。

上記の研究のように極性情報を考慮して分散表現の学習を行うことで、異なる極性を持つ語を区別してベクトル空間に埋込むことができる。

表1に実験に用いた学習モデルにおける感情語のcos類似度を示す。表1における細線の感情語は肯定の極性を、太線の感情語は否定の極性を表す。表1から極性情報を考慮しないモデルと比べて極性情報を考慮するモデルの方が同じ極性の感情語は類似性が高く、異

表 1 感情語の cos 類似度

感情語対	極性情報あり		極性情報なし	
	SE	StarSpace	Word2Vec	GloVe
美味しい vs. 不味い	-0.78	-0.78	0.73	0.82
親切 vs. 不親切	-0.77	-0.81	0.29	0.45
不愛想 vs. ぶっきらぼう	0.91	0.89	0.83	0.89

なる極性の感情語の類似性は低くなるように学習が行われていることがわかる。

### 3 極性情報を考慮した分散表現学習モデル

この節では実験に用いた極性情報を考慮した学習モデルについて説明する。

#### 3.1 Sentiment embedding

Sentiment embedding(SE) モデル [13] は対象単語の局所的な文脈と対象単語の含まれる文の評価極性ラベルを利用して分散表現を学習する。

SE モデルは対象単語  $w_t$  と前後  $C$  個の文脈語から成る単語列  $x = (w_{t-C}, \dots, w_t, \dots, w_{t+C})$  を入力として学習を行う。モデルはまず単語列  $x$  を埋込み層でベクトル表現に変換し、それぞれのベクトルを連結することでベクトル  $O_{lookup} \in R^{1 \times (n \times d)}$  を作成する。ここで  $n$  は単語列  $x$  に含まれている単語数、 $d$  は埋込み層の次元数を表す。次にベクトル  $O_{lookup}$  は式 (1)(2) の線形変換層によって線形変換され出力  $f_c(x)$  を得る

$$O_1 = \sigma(W_1 O_{lookup} + b_1) \quad (1)$$

$$f_c(x) = W_c O_1 + b_c \quad (2)$$

ここで、 $W_1$  および  $W_c$  は重み行列を表し、 $b_1$  と  $b_c$  はバイアス項を表す。また、 $\sigma$  はベクトルの要素ごとに非線形変換を行う tanh 関数である。

対象単語と文脈語の関連性の推定値である  $f_c(x)$  は次の式 (3) によって最適化される。

$$loss_c = \max(0, 1 - f_c(x) + f_c(x')) \quad (3)$$

式 (3) における  $x'$  は単語列  $x$  内の対象単語  $w_t$  を語彙  $V$  内のランダムな単語に置き換えることで作成したノイズを含む単語列である。出力  $f_c(x')$  はこのノイズを含む単語列を入力とした場合のモデルの出力であり、出力  $f_c(x')$  と正しい単語列に対する出力  $f_c(x)$  との-margin が少なくとも 1 となるように最適化することで対象単語とその文脈語の関連性を学習する。

加えて、式 (1) の出力ベクトル  $O_1$  を次の式 (4) を用いて線形変換することで文の評価極性の推定値である

$f_s(x)$  を得る。また、出力  $f_s(x)$  は次の式 (5) によって最適化される。

$$f_s(x) = W_s O_1 + b_s \quad (4)$$

$$loss_s = \max(0, 1 - \delta(t)f_{s0}(x) + \delta(t)f_{s1}(x)) \quad (5)$$

式 (5) における  $f_{s0}(x)$  は否定のラベルの推定値を、 $f_{s1}(x)$  は肯定のラベルの推定値を表す。また、関数  $\delta$  は文の評価極性の正解ラベル  $t$  を入力として正解ラベルが 1 の場合は 1 を、0 の場合は  $-1$  を返す関数である。

式 (5) によって肯定/否定ラベルの推定値の-margin を最適化することで、対象単語とその文脈語がレビュー文の評価極性とどのように関係しているかを学習する。

以上で得られた 2 つの損失誤差を式 (6) を用いて組み合わせることでモデル全体の損失誤差とする。

$$loss_{all} = \alpha loss_s + (1 - \alpha) loss_c \quad (6)$$

式 (6) 中の  $0 < \alpha < 1$  は全体の損失誤差における 2 つの誤差損失の重みを示すハイパーパラメータである。

#### 3.2 StarSpace

StarSpace[14] は単語や文章あるいは商品推薦におけるユーザの購入商品などの様々な対象の分散表現を学習できる汎用的な学習モデルである。学習対象は離散的な特徴集合 (Bag-of-features) で表現され、それらの分散表現の類似性を比較することで対象間の類似点を学習する。本実験では、学習対象は各レビュー文  $S$  とその極性ラベル  $L$  となる。

レビュー文  $S$  は BoW (Bag-of-Words) 表現で表され、モデルの埋込み層によって  $d$  次元のベクトルに埋込まれる。すなわち、レビュー文  $S$  に含まれている単語  $w_i$  を要素として単語ごとの分散表現  $e_{w_i}$  を獲得し、そのベクトルの総和  $e_S = \sum e_{w_i}$  をレビュー文の分散表現とする。また、極性ラベル  $L$  のラベル自身を固有の要素として分散表現  $e_L$  を獲得する。この分散表現の類似性を比較することでモデルは学習を行う。具体的には、次の損失関数を最小化する。

$$F(\text{sim}(e_S, e_L), \text{sim}(e_S, e_{L-1}), \dots, \text{sim}(e_S, e_{L-N})) \quad (7)$$

ここで、関数  $\text{sim}$  は与えられたベクトルの類似性を計算する関数であり、ベクトル間の cos 類似度あるいは内積を計算する。また、 $e_{L-}$  は対象のレビュー文の極性ラベルとは異なる極性ラベルの分散表現である。損失関数  $F$  はレビュー文と正しい極性ラベルとの類似性と異なった極性ラベルとの類似性を比較する関数であ

り, Hinge 損失あるいは softmax 関数による負の対数損失を計算する.

今回の実験では, 類似性計算に  $\cos$  類似度, 損失関数  $F$  には Hinge 損失を用いた.

## 4 実験

次に示す 3 つのニューラルネットワークを用いた極性分類モデルと 4 種類の単語分散表現を利用してレビュー文の評価極性分類を行う.

### 4.1 評価極性分類モデル

#### 4.1.1 Deep average network (DAN)

多層順伝播ニューラルネットワークを用いた分類モデル. 文に含まれている単語の分散表現の平均を入力とする. モデルを多層化することによって入力から文の意味表現の違いを学習する. 加えて, 単語ごとの分散表現の平均によって表現される文の分散表現の一部(文内の単語)をランダムに削除することで, 学習の際のユニークな学習例を増加させる [4]. 今回の実験では 3 層の中間層を持つモデルを構築した.

#### 4.1.2 Recursive neural network (RNN)

文の構文木を用いた分類モデル. 文に含まれている単語の分散表現から句の分散表現を構成し, これを再帰的に繰り返すことで文の分散表現を作成して分類を行う. 構文木の情報を利用することで構文的に意味のある文の分散表現を作成することができる [9][10].

#### 4.1.3 Tree-LSTM

RNN に LSTM(Long Short-Term Memory)[3] 構造を適用することで, 長い系列データを入力とした場合の勾配損失の問題に対処した分類モデルである [11].

### 4.2 初期化に用いる単語分散表現

埋込み層の初期値には 3 節で説明した極性情報を考慮した分散表現学習モデルに加えて, 比較対象として極性情報を考慮せず学習を行うモデルである Word2Vec(Skip-gram) および GloVe[8] の 2 つのモデルを利用する.

各分散表現モデルは楽天データ<sup>1</sup>の楽天トラベル: レビューデータにおける 1996 年から 2007 年までの 150 万レビューを用いて学習を行う.

極性情報を考慮したモデルの学習には各レビュー文ごとに極性ラベルを付与する必要がある. Tang らの先行研究では Twitter の Tweet の集合を学習対象として Tweet に含まれる顔文字を利用して極性ラベルの自動

ラベリングを行っている. この様に学習データが自動でラベリングされた場合でも, 相当数のデータを収集することで単語と文の極性ラベルの関係を学習できるとされている.

本実験では, 既存の感情語辞書<sup>2</sup>を用いてレビュー文の自動ラベリングを行う. 具体的には, 各レビューを文ごとに分割し, 分割されたレビュー文に含まれている感情語がすべて同じ極性ラベルの場合にその文に感情語と同じラベルを付与する. 異なるラベルの感情語が含まれる文については学習データから除外する. 加えて, 否定表現を伴う感情語について考慮するために [17][18] を参考にして否定表現の出現パターンを作成して評価極性の反転に対応する.

以上の方法でレビューを文ごとに分割して自動ラベリングを行うことで 280 万文のレビュー文を収集し, このレビュー文集合を学習データとして利用する.

全てのモデルの分散表現の次元数は 100 とし, 考慮する前後の文脈語の数は 5 とした. 負例のサンプル数は Word2Vec と GloVe および StarSpace は 5, SE モデルは 100 である. また, SE モデルの中間層の次元数は 100 とし, SE モデルの損失誤差の結合パラメータ  $\alpha$  は 0.6 とする.

### 4.3 実験設定

分類精度評価のために楽天データにおける筑波大学文単位評価極性タグ付きコーパス (TSUKUBA コーパス) を利用する. TSUKUBA コーパスは楽天トラベルにおける 1000 件のレビューを対象として文単位で評価極性情報を付与したコーパスである. このコーパスは 2 人の作業者によって褒め・苦情・要求・ニュートラルの 4 種類の評価極性タグが付与されているが, 今回の実験では苦情・要求を 1 つの分類ラベルとして 3 値の分類を行う. また, 2 人の作業者によって同じ極性タグが付与され, かつ, 文内のすべての単語が 3 節で学習した分散表現モデルの語彙に含まれている 3593 文を分類対象とする.

RNN および Tree-LSTM で利用する構文木については CaboCha[5] を用いて各レビュー文を係り受け解析することで作成する. 加えて, 4.2 節で用いた感情語および否定表現を伴った感情語が文内に出現した場合にその単語や句に対応する極性を教師ラベルとして利用する.

実験は 10 分割交差検定を用いて行う. 各モデルは評

<sup>1</sup> [https://rit.rakuten.co.jp/data\\_release\\_ja/](https://rit.rakuten.co.jp/data_release_ja/)

<sup>2</sup> 日本語評価極性辞書 [15][16]

価尺度として F 値を用いる。

表 2 に各モデルにおける評価極性分類の結果を示す。表 2 の結果から、すべての分類モデルで埋込み層の重みをランダムな値で初期化する場合と比べて、事前学習された分散表現を用いて初期化する場合の精度が高いことがわかる。このことから、埋込み層の重みを分散表現を用いて初期化することが分類精度の向上につながることを確認できる。しかし、すべての分類モデルで Word2Vec を用いた場合が最も高い精度となった。SE と GloVe の間には大きな精度の差は見られなかったが、すべての分類モデルにおいて StarSpace を用いた場合が最も低い精度を示した。

文脈情報のみから学習を行う Word2Vec と比べて SE および StarSpace の極性情報を用いた分散表現学習モデルの精度が低くなってしまった原因として、極性を持たない『ホテル』などの名詞、あるいは助詞などの単語の類似性・関連性を正しく学習できていないことが考えられる。特に、StarSpace は対象語と前後の文脈語の関連性を考慮せずに学習を行うため最も低い精度となったと考察する。

## 5 おわりに

本研究では、分類モデルの埋込み層の初期値として極性情報を考慮した単語分散表現を用いることの有用性を検証した。実験の結果、Word2Vec などの極性情報を考慮せず学習を行うモデルと比べて低い分類精度となり、有用性は確認できなかった。しかし、極性情報を考慮した分散表現は極性を持つ単語の類似性を正しく学習できる点において、感情語辞書の登録語と未登録語の類似性を判断することで辞書の拡張などに応用することができると考える。今後は、極性情報を考慮した分散表現を用いた辞書や学習データの拡張について検討していきたい。また、文単位の分散表現を学習モデルの入力として利用した場合や他の付加情報を用いた学習モデルについても検証していきたい。

## 謝辞

分類実験を行うにあたり、楽天データにおける楽天トラベル: レビューデータ及び極性タグ付きコーパスを使用させて頂きました。関係者に深く感謝いたします。

## 参考文献

[1] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks

表 2 各モデルにおける F 値

	F 値		
	DAN	RNN	Tree-LSTM
SE	84.1	81.9	84.0
StarSpace	82.8	80.5	82.1
Word2Vec	<b>85.8</b>	<b>82.7</b>	<b>87.4</b>
GloVe	84.2	81.9	86.0
Random	79.3	74.5	81.9

with multitask learning. In ACM, 2008.

[2] Zellig S. Harris. Distributional structure. Word 10.2-3 : 146-162, 1954.

[3] Sepp Hochreiter, and Jrgen Schmidhuber. Long short-term memory. Neural computation 9.8 : pp. 1735-1780, 1997.

[4] Mohit Iyyer, Varun Manjunatha, Jordan BoydGraber, and Hal Daum'e III. Deep unordered composition rivals syntactic methods for text classification. In ACL, 2015.

[5] Kudo, Taku, Yuji Matsumoto. Fast methods for kernel-based text analysis. In ACL, 2003.

[6] Igor Labutov, Hod Lipson. Re-embedding words. In ACL, 2013.

[7] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In ICLR, 2013.

[8] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In EMNLP, 2014.

[9] Richard Socher, et al. Semi-supervised recursive autoencoders for predicting sentiment distributions. In EMNLP, pp. 151161, 2011.

[10] Richard Socher, et al. Recursive deep models for semantic compositionality over a sentiment treebank. In EMNLP, pp. 16311642, 2013.

[11] Kai Sheng Tai, Richard Socher, and Christopher D. Manning. Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks. In ACL. pp. 1556-1566, 2015.

[12] Duyu Tang, et al. Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification. In ACL, 2014.

[13] Duyu Tang, et al. Sentiment embeddings with applications to sentiment analysis. IEEE Transactions on Knowledge and Data Engineering 28.2 : pp. 496-509, 2016.

[14] Ledell Wu, et al. StarSpace: Embed All The Things!. arXivpreprint arXiv:1709.03856, 2017.

[15] 小林のぞみ, 乾健太郎, 松本裕治, 立石健二, 福島俊一. 意見抽出のための評価表現の収集. 自然言語処理, Vol.12, No.3, pp.203-222, 2005.

[16] 東山昌彦, 乾健太郎, 松本裕治. 述語の選択選好性に着目した名詞評価極性の獲得. 言語処理学会第 14 回年次大会論文集, pp.584-587, 2008.

[17] 松吉俊. 否定の焦点情報アノテーション. 自然言語処理 21.2 : 249-270, 2014.

[18] 日本語記述文法研究会. 現代日本語文法 3. くろしお出版, 2007.