

# 単語分散表現を用いた 多層 Denoising Auto-Encoder による評価極性分類

張 培楠<sup>1,a)</sup> 小町 守<sup>1,b)</sup>

## 概要：

年々増加し続けている莫大なウェブデータからのマイニングなど、評価極性分類タスクは重要な役割を担っている。しかし、評価極性辞書を用いたりモデルや素性を設計・チューニングして機械学習させる従来のアプローチは専門家の高度な知識に依存しており、一般化できない点が課題であった。たとえば Nakagawa ら [1] は係り受け関係といった構文木情報を参照して 12 種類もの素性テンプレートを作成し、複雑なモデルを設計していた。また、Choi [2] らのように言語に依存したルールや、Ikeda [3] らのように分類に極性辞書を必要とすることで日々増加する新語や未知語に対応できない問題が挙げられる。そこで、本研究では単語の分散表現と深層学習のいち手法である多層 Denoising Auto-Encoder を用いて、辞書のような語彙知識や複雑なモデル設計、そして素性エンジニアリングをしなくても高い性能が得られる評価極性分類手法を提案する。

## 1. はじめに

インターネットが普及し、テキストによる評価情報が注目を集めている。また、ソーシャルメディアやブログといったユーザ発信サービスの利用率は年々上昇し、特定のショッピングサイトに記述されていない商品やサービスに関する莫大な極性情報を含んでいる。そのため、こういった極性を含む評価（肯定・否定）情報を適切に分類することは必要不可欠な技術といえる。

このような評価極性分類タスクにおいての一般的な目標は、ある文もしくは文章を与え、それをポジティブ（望ましい）とネガティブ（望ましくない）に分類することである。その際に機械学習で用いられる最も基礎的な素性は単語（Bag-of-Words）素性であり [4,5]、例えば「便利」という単語にはポジティブに分類される重みが学習されやすく、「不便」という単語ネガティブに分類される重みが学習されやすい。

しかし、単語素性のみを使った分類では限界がある。まず、従来のヒューリスティックに分類ルールを作成する手法や極性辞書を用いた手法は言語に依存していたり、極性辞書を作る必要が人的コストが大きい。さらに日々増加する新語や未知語に対応することが困難であることが予想される。したがってデータスパースネス問題が引き起こ

される可能性があるため、表層の素性だけでは不十分といえる。

また、統語構造を考えないため「良いデザインだけど不便」と「不便だけど良いデザイン」が同じ極性になってしまうが、主節が主な意見情報を担っていることを考慮すると、前者はややネガティブ、後者はややポジティブな意見情報を表現していると考えられるが、このような違いを Bag-of-Words 素性では表現できない。

この統語構造の問題を解決するため、Nakagawa ら [1] は係り受け関係を用いて構文木を作成し、その依存部分木に極性を割り当てる手法を提案した。しかしそれには複雑な係り受けに基づく素性テンプレートを設計するため、専門的知識が必要かつモデルが複雑になってしまう。

そこで我々は極性辞書のような語彙知識を用いなくても単語の意味を表現でき、かつデータスパースネスに頑健な分散表現と、複雑な素性やモデル設計を必要とせず、かつ高い表現力を持つ深層ニューラルネットワークに注目した。したがって本研究では単語の分散表現と多層 Denoising Auto-Encoder（以下 SdA と略す）を用いて評価極性分類タスクに取り組む。

単語の分散表現では、単語をベクトルとして表現する。しかし分散表現では単語ベクトルを従来の 1-of-K のような疎なベクトルではなく、単語自体の意味を表す密なベクトルとして扱う。つまり意味が近い単語同士のベクトルの距離は近いということになる。極性分類タスクにおいて単

<sup>1</sup> 首都大学東京 システムデザイン研究科

<sup>a)</sup> zhang-peinan@ed.tmu.ac.jp

<sup>b)</sup> komachi@tmu.ac.jp

語の意味は極性を大きく左右するため、記号的な意味しかない 1-of-K 表現よりも、より意味を捉えることができる分散表現の方が望ましい。その単語分散表現を学習する手法として Mikolov ら [6,7] の Skip-gram や CBOW のモデルが近年大きな成果を収めている。

SdA は Denoising Auto-Encoder (以下 dA と略す) を多層に重ねたもので、高い汎化能力を持ったニューラルネットワークである。積み重ねられた dA によって抽出される特徴は層が深くなっていくのに連れてより抽象的な表現を獲得できる。この手法は音声認識 [8] や画像処理 [9,10], 分野適応 [11] など高い表現能力を証明している。

本研究の主要な貢献は以下の 2 点である。

- 評価極性分類タスクにおいて、大規模コーパスから学習した単語分散表現を用いることと、多層 (3 層以上) の SdA を用いることが分類性能に大きく寄与することを示した。
- 日本語の評価極性分類タスクに対し、複雑なモデルを設計することなく、現時点における世界最高精度を達成した。

## 2. 関連研究

ここでは本研究に関連する研究を、評価極性分類タスクと深層学習 (単語の分散表現・多層ニューラルネットワーク) に分け説明する。

### 2.1 評価極性分類

評価極性分類タスクについてはこれまでに多くの研究がなされてきた。

Nakagawa ら [1] は、評価表現の依存構造木を考え、個々の部分依存構造木に対する評価極性を隠れ変数で表し、隠れ変数間の相互作用を考慮した日本語と英語の評価極性分類を条件付き確率場 (CRF) [12] で行った。これにより、評価極性の反転などの統語構造を考慮することを可能とした。しかし、12 種という非常に多くの素性テンプレートを作る必要があり、複雑なモデルである。それに対して、本研究での分類手法は素性エンジニアリングする必要がないため、比較的簡単なモデルである。

同じことが Choi ら [2] にも言える。Choi らは、係り受け関係を考慮した手法で評価極性の分類を行っているが、単語単位の極性からは人手によってルールを作成し、評価表現全体の極性を構成している。また、これらのルールは英語に基づいているため、日本語での分類を考えた際には新たに日本語のルールを作成しなければならない。しかし本研究では文の分散表現を単語の分散表現の平均で計算しているため言語に依存しない手法である。

Ikeda ら [3] は構成する単語の周辺を見ることで評価極性を学習する手法で文分類を行ったが、評価表現辞書を使うため、辞書に登録されていない単語にはうまく対処でき

ない。加えて、辞書を作成する必要があり、人的リソースがかかる上新語に迅速に対応できない。一方、本研究は極性辞書を用いないので未知語があっても大規模な言語データがあれば頑健に動作する。

### 2.2 深層学習

深層学習の大きな特徴のひとつとして、内部に素性抽出器を持つことで従来のように素性を人手によって設計する必要がないことがあげられる。これにより、素性エンジニアリングでの時間や労力を節約できる他、人間では作り得ないような有効な素性を取り入れることができるようになる。

Collobert ら [13] は入力である任意の長さの文を固定次元に変換するために、単語ベクトルを 3 つずつ畳み込むことで局地的な特徴を得たのち、各次元の最大値を取得して新たなベクトルにプーリングした。このプーリングされたベクトルを従来のニューラルネットワークを用いて単語分散表現の表現学習を行った。これは本研究で使用している表現学習とは異なる手法で単語分散表現を作成している。

Glorot ら [14] は分野適応タスクに SdA を使用した。Amazon の 4 つの分野の商品レビューに対する評価を、それぞれ学習させた学習器を別の分野でテストを行い、その性能を測った。この研究には感情分類タスクも含まれているが、入力に Collobert ら [13] の手法で表現学習した最もよく出現する動詞のみを使用するため、文ベクトルを作成する本研究とは異なる。

Collobert ら [13] や Mikolov ら [6,7] の表現学習を用いて単語の分散表現を作成し、Twitter などのマイクロブログの極性分類を行った研究として、dos Santos ら [15] と Tang ら [16] が挙げられる。両者とも感情分類タスクであるが、前者は文ベクトルを文字単位の畳み込みネットワークで生成したのに対し、後者は単語の分散表現自体を極性として表現したモデルを使用した。本研究では単語の分散表現を用いて単語単位のベクトルで文ベクトルを生成するためこれらの研究とは異なる。

Socher ら [17,18] は共通の Auto-Encoder を再帰的に配置し、入力するベクトルを連結 (concatenate) させることで単語の順番といった構文的な情報を考慮したモデルを作成した。また、構文木に AE を配置することで、フレーズ単位の極性を表現することを可能とした。この再帰的なニューラルネットワークを使用して文ベクトルを作成したが、これは本研究での文ベクトルを作成する手法とは異なる。

## 3. 単語分散表現を用いた多層 Denoising Auto-Encoder による評価極性分類

本研究では、日本語の文が与えられた時に、その極性をポジティブあるいはネガティブの 2 値に分類するタスクを

考える。

提案手法は入力された文から文ベクトルを作成し、学習器への入力とする。文ベクトルは文に含まれる単語ベクトルの平均によって得る。単語ベクトルは分散表現を用いる。

3.1 節では分散表現、3.2 節では多層ニューラルネットワークの説明をしていく。

### 3.1 分散表現

伝統的な単語のベクトル表現の方法に、Bag-of-Words を作る時に用いる 1-of-K 表現がある。1-of-K 表現による単語ベクトルの次元は語彙数と等しく、各次元が一つの単語に対応づく。1-of-K 表現では表層的に異なった単語同士は全く別の記号として扱うが、例えば「犬」と「猫」のようにどちらもペットとして飼われる動物であれば、似たような文脈で使用されることから、同じようなベクトルで表現したい。それを解決するのが分散表現である。

分散表現を学習するタスクは表現学習と呼ばれ、言語の意味的な類似度を捉える必要のある問題で近年注目を集めている。単語の表現学習では、各単語に低次元で密な実数値ベクトルを割り当て、各単語ベクトル表現がその単語の特徴を捉えるように大量のテキストコーパスからベクトル空間を学習する [19]。

#### 3.1.1 単語の分散表現

出現する語彙の数を  $|V|$ 、単語を表現するベクトルの次元を  $d$ 、1-of-K ベクトルを  $\mathbf{b} \in \mathbb{R}^{|V|}$ 、単語ベクトルの集合の行列を  $\mathbf{L} \in \mathbb{R}^{d \times |V|}$  とすると、ターゲットとしている  $k$  番目の単語  $w_k$  のベクトルは式 1 のように表すことができる。

$$\mathbf{w}_k = \mathbf{L}\mathbf{b}_k \quad (1)$$

単語の分散表現の学習には 2.2 節で挙げた Collobert ら [13] の手法以外に、word2vec で使用する Continuous Bag-of-Words (CBOW) もしくは Skip-gram が使用される手法もある [6,7]。前者は単語周辺の文脈から決めたウィンドウサイズ分の単語を使用して注目している単語を推定するモデルで、後者は CBOW とは逆に注目している単語からウィンドウサイズ分の周辺の単語を推定するモデルである。Mikolov らの研究では Skip-gram の方がよりよい性能を発揮したと示している。そのため本研究では Skip-gram を使用した。

#### 3.1.2 文の分散表現

本研究では、コーパスにある  $|M|$  個の文を用いて、本タスクで使用する文行列  $\mathbf{S} \in \mathbb{R}^{|M| \times d}$  を作成する。

まず単語ベクトルを用いて文ベクトルを作成する。 $|N|$  単語から構成される  $1 \leq i \leq M$  番目の文が与えられた時に、その文に含まれている単語の特徴ベクトルから文ベクトル  $\mathbf{S}^{(i)} \in \mathbb{R}^d$  を作成する。この時、文ベクトル  $\mathbf{S}^{(i)}$  の  $1 \leq j \leq d$  番目の要素  $\mathbf{S}_j^{(i)}$  は式 2 のように average で算出

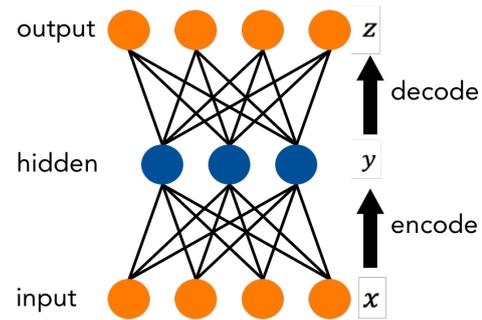


図 1 Auto-Encoder

する [20]。

$$\mathbf{S}_j^{(i)} = \frac{1}{N} \sum_{n=1}^N \mathbf{w}_n^{(i)} \quad (2)$$

次に、求める文行列  $\mathbf{S}$  を式 3 のように算出する。

$$\mathbf{S} = \begin{bmatrix} \mathbf{S}^{(1)T} \\ \mathbf{S}^{(2)T} \\ \vdots \\ \mathbf{S}^{(M)T} \end{bmatrix} \quad (3)$$

### 3.2 Auto-Encoder

ニューラルネットワークにおいて、層がより深くなっていくのに連れて、誤差逆伝搬で学習させるには時間がかかる上に過学習もしやすくなる傾向がある。したがって、各層で事前学習によって特徴抽出しやすい初期値を持たせるために Auto-Encoder を用いる手法が提案されている。

Auto-Encoder は、Hinton ら [21] によって考案された教師なし学習のいち手法で、ニューラルネットワークを使用して入力の次元を中間層で圧縮するアルゴリズムからなっている。これは入力と同じになるような出力の中間層での表現を学習し、その中間層の次元は入出力のものより小さくすることで次元圧縮を可能としている (図 1)。このとき、入力を中間層にエンコードする式を式 4、中間層から出力層にデコードする式を式 5 とする。

$$\mathbf{y} = s(\mathbf{W}\mathbf{x} + \mathbf{b}) \quad (4)$$

$$\mathbf{z} = s(\mathbf{W}'\mathbf{y} + \mathbf{b}') \quad (5)$$

このとき、 $s(*)$  は tanh やシグモイド関数で、 $\mathbf{W}, \mathbf{W}'$  は重み行列、 $\mathbf{b}, \mathbf{b}'$  はバイアス項を表している。

$d$  次元の入力ベクトル  $\mathbf{x}$  と出力ベクトル  $\mathbf{z}$  の差を表す損失関数は、式 6 のように交差エントロピー関数で表されており、これを確率的勾配降下法 (SGD) や AdaDelta などの最適化手法により最小化していく。

$$L_H(\mathbf{x}, \mathbf{z}) = - \sum_{k=1}^d [\mathbf{x}_k \log \mathbf{z}_k + (1 - \mathbf{x}_k) \log(1 - \mathbf{z}_k)] \quad (6)$$

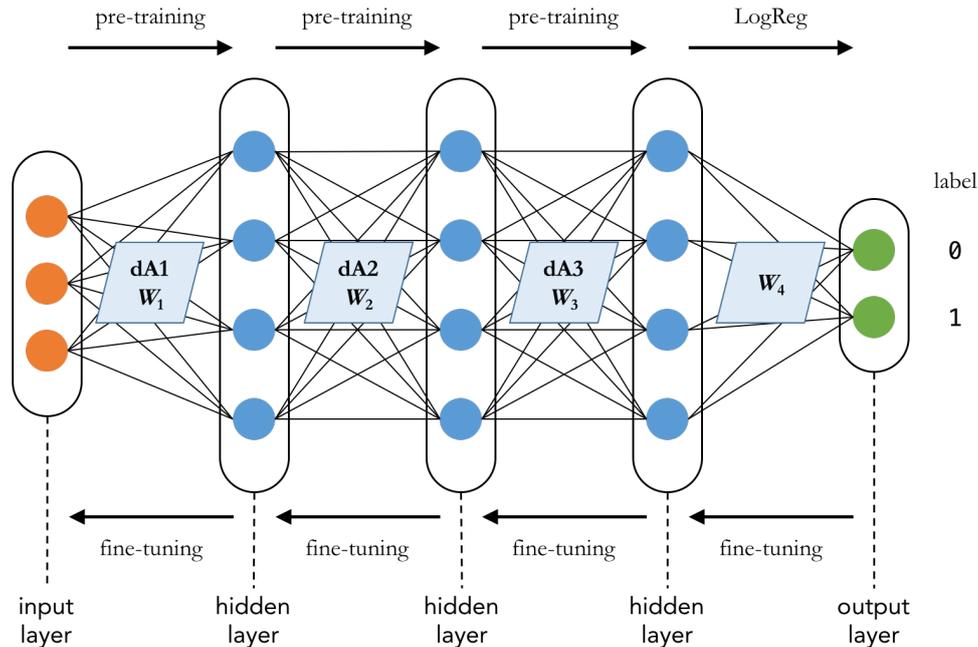


図 2 Stacked Denoising Auto-Encoder

### 3.2.1 Denoising Auto-Encoder

従来の多層パーセプトロンでは損失関数に正則化項を使用することがほとんどだが, Auto-Encoder では Denoising を使用することで正則化と同等の役割を果たすことができる.

Denoising Auto-Encoder は Auto-Encoder の確率的なバージョンで, 入力に対して学習のたびにランダムなノイズを与えて事前学習することで, より高い汎化能力を持った初期値を決定することができる手法である. 損失関数はノイズの入っていない入力を評価するため, Denoising を用いた Auto-Encoder ではより良い特徴を抽出することが可能である. [22] 似たような正則化の働きをする手法として Dropout [23] があるが, これは入力ではなく中間ノードを一定の確率で無視して学習を行う. これにより各データごとに異なるモデルを使用して学習し, 推定時にはあらかも全部の結果の平均を使っていることに対応しているので過学習しにくくなる.

### 3.2.2 Stacked Denoising Auto-Encoder

dA を多層に積み重ねることで, より表現能力を向上させた手法が Stacked Denoising Auto-Encoder である. 積み重ねられた dA によって抽出される特徴は層が深くなっていくのに連れてより抽象的な表現を獲得できる. [24]

SdA による学習は 2 つの段階からなっている. まず, ニューラルネットワーク各層を dA として教師なしの事前学習 (pre-training) を行い, 次にニューラルネットワーク全体で教師ありの微調整 (fine-tuning) を行う. 事前学習では, 入力  $A_i$  に始まり各層の dA による特徴抽出がなされ, 抽出された内部表現である中間層が次の dA の入力と

して渡される. 最後の中間層まで事前学習すると, その中間層がロジスティック回帰によって分類され, 分類結果のベクトルが出力層に渡される. 微調整では出力層から正解ラベルを使い誤差逆伝搬により各層での重み行列の更新がなされる. (図 2)

図 2 において, 入力層は式 2 で得られる  $S_i \in \mathbb{R}^d$  で, 1 層目で重み行列  $W_1$  を持つ dA1 によって最初の中間層に変換される. なお, 中間層の数と中間ノードの次元はハイパーパラメータになっており, ここでは  $i$  層目の中間ノード次元を  $n_i$  とする. したがって  $W_1$  の次元は式 4 から  $n_1 \times d$  となる. 同様に,  $l-1$  層目までの重み行列は  $W_i \in \mathbb{R}^{n_i \times n_{i-1}}$  となる (ただし  $i > 2$ ). 最後の  $l$  層では, ロジスティック回帰によって正解ラベルと同じ次元  $d_{ans}$  の出力をするため, 重み行列  $W_l$  の次元は  $d_{ans} \times n_{l-1}$  となる.

## 4. 評価極性分類実験

### 4.1 比較手法

SdA の有効性を示すために, 線形分類器 (ロジスティック回帰)\*1 による分類との比較を行った. (LogRes-w2v)

単語の分散表現の有効性を示すために, Bag-of-Features を使った手法との比較を行った. (LogRes-BoF, SdA-BoF) 1-of-K 行列で式 2 と同じ方法で文ベクトル  $S \in \mathbb{R}^{|V|}$  を作成し, それらに対して主成分分析 (PCA)\*2 で  $d = 200$  次元まで次元削減を行った.

ここでは同じコーパスを使用した Nakagawa らの手法と

\*1 SdA 及びロジスティック回帰はともに Theano version-0.6.0 を用いて実装した.

\*2 scikit-learn version-0.10 を使用した.

本研究で示した手法、そして比較するハイパーパラメータの説明をする。

#### Tree-規則

評価表現の部分依存構造木の評価極性を、規則により決定的に決めていく方法。[1]

**Tree-CRF** 隠れ変数を持つ CRF による手法。[1]

**MFS** データセットの中でもっとも大きい割合の回答（ここではネガティブ）を常に選ぶベースライン。

**LogRes-BoF** 1-of-K の単語ベクトルの average で文ベクトルを作成し、PCA により 200 次元まで次元削減を施した行列を線形分類器（ロジスティック回帰）で分類する。

**SdA-BoF** LogRes-BoF で作成したベクトルを SdA で分類する。

**LogRes-w2v** 単語の分散表現で作成した文ベクトルを、線形分類器（ロジスティック回帰）で分類する。

**SdA-w2v** 3 節で説明した、単語の分散表現で作成した文ベクトルを SdA を用いて分類を行う提案手法。

**SdA-w2v-neg** Nakagawa らと同様、否定表現に対して処理を施して単語の分散表現を作成し、SdA を使って分類を行う。

また、提案手法に関するハイパーパラメータの Denoising と中間層数、中間ノード次元数は以下のように調節した。

Denoising の有効性を示すために、Denoising を施していない手法（ノイズレートの 0%）と Denoising を行った手法（ノイズレートが 10%、20%、30%）のものと比較を行った。Denoising の手法として、指定されたノイズレートで 0 を含む二項分布で乱数を生成し、それを入力ベクトルの対応する要素にかける。

多層の有効性を示すために、中間層の数が 1 層のもの、2 層以上のものとの比較を行った。

ネットワークの表現力を調べるために、中間ノードの次元数を 100, 300, 500, 700 次元で比較を行った。

#### 4.2 使用データ・ツール

単語の分散表現の作成にあたって、Mikolov ら [6, 7] の Skip-gram から作成される word2vec<sup>\*3</sup> を使用した。学習には日本語版 Wikipedia のダンプデータ（2014 年 11 月時点）を使い、MeCab<sup>\*4</sup> にて分かち書きしたのち word2vec を用いてサイズ 200 次元で単語の分散表現を作成した。できあがったモデルの語彙数は、否定表現を処理しなかった手法では 426,782 語、否定表現を処理した手法では 431,456 語である。

実験には NTCIR-6 の意見分析パイロットタスクテストコレクション [25] を使用した。NTCIR-6 コーパスは大規模なテストコレクションを用いた検索、アジア言語と英語

表 1 評価極性分類精度

手法	正解率
Tree-規則 [26]	72.6 %
Tree-CRF [26]	78.9 %
MFS	70.4 %
LogRes-BoF	70.8 %
SdA-BoF	76.9 %
LogRes-w2v	79.5 %
<b>SdA-w2v</b>	<b>81.7 %</b>
SdA-w2v-neg	80.9 %

の言語横断検索、質問応答、言語横断情報アクセスに焦点を当てた NTCIR-6 ワークショップで作成されたコーパスである。本研究で使用した日本語のデータでは、毎日新聞と読売新聞の記事に対して 3 人のアノテータにより文単位で評価極性タグが付けられている。本研究では、同じ文に対して 3 人のアノテータのアノテーション結果の和集合を取ったものを使用した。アノテータのタグがポジティブとネガティブの両方を含む場合、常に特定のアノテータのタグを使用する。その合計は 2,599 文で、ポジティブが 769 文、ネガティブが 1,830 文である。NTCIR-6 コーパスにはニュートラル極性も含まれているが、本実験では二値分類であるためニュートラル極性は取り除いた。

また実験は 10 分割交差検定により行い、性能は正解率 (Accuracy) によって評価した。

#### 4.3 実験結果

ここでは上で挙げた手法での実験結果と、パラメータチューニングの比較をしていく。

まず、各手法を用いた NTCIR-6 での正解率を表 1 に示す。

Nakagawa ら [1] の提案手法である Tree-規則と Tree-CRF の正解率はそれぞれ 72.6% と 78.9% であった。本研究においてのベースラインの MFS の正解率は 70.4% で、Bag-of-Features を使用したロジスティック回帰と SdA による分類器の正解率はそれぞれ 70.8% と 76.9% であった。同じように、単語の分散表現を使ったロジスティック回帰と SdA による分類の正解率はそれぞれ 79.5% と 81.7% で、SdA-w2v と同じパラメータで、否定表現処理を施したデータに対して分類したところ、正解率が 80.9% になった。なお、SdA での実験では、後述するパラメータチューニングを行って最も正解率が高かった組み合わせ（中間層 4 層、中間ノード 500 次元、ノイズレート 10%）を使用している。<sup>\*5</sup>

また、ハイパーパラメータごとの比較を表 2 のように行った。

表 2 上段では、中間層を 3 層、中間ノードを 100 次元で

<sup>\*3</sup> <https://code.google.com/p/word2vec/>

<sup>\*4</sup> MeCab version-0.996, IPADic version-2.7.0

<sup>\*5</sup> 開発データは使用せず、10 分割交差検定による正解率で評価した。

表 2 ハイパーパラメータによる精度変化

パラメータ		正解率
ノイズレート	0 %	81.1 %
	10 %	<b>81.5 %</b>
	20 %	81.4 %
	30 %	80.9 %
中間層数	1	80.6 %
	2	80.4 %
	3	81.1 %
	4	<b>81.6 %</b>
	5	81.4 %
	6	81.1 %
中間ノード次元	100	81.1 %
	300	81.2 %
	500	<b>81.3 %</b>
	700	81.2 %

固定して各層のノイズレートの変化に伴う正解率の変化を比較した。ノイズレートが 10 % の時にもっとも高い正解率が得られた。

表 2 中段では、中間ノードを 100 次元、ノイズレートを 0 % に固定して、層の数を変化させて正解率の変化を比較した。中間層が 4 層の時にもっとも高い正解率が得られた。

表 2 下段では、中間層を 3 層、ノイズレートを 0 % に固定して、中間ノードの次元を変化させて正解率を算出した。中間ノードが 500 次元の時にもっとも高い正解率が得られた。

## 5. 考察

実験により得られた結果 (表 1) とパラメータチューニングの結果 (表 2), 及び文の例 (表 3) を元に考察を行う。

### 5.1 手法

**Bag-of-Features vs. 単語分散表現** 学習手法を線形分類器 (ロジスティック回帰) で固定した場合、正解率は Bag-of-Features では 70.8 % で単語分散表現では 79.5 % であった。SdA で学習した場合、Bag-of-Features は 76.9 % で単語分散表現は 81.7 % であった。これらの結果から、Bag-of-Features よりも単語分散表現を使用した方が正解率は 4.8 ~ 8.7 ポイントもの向上が見られた。このことから、このタスクにおいての単語分散表現の貢献は非常に大きいといえる。

**線形分類器 vs. 多層 Denoising Auto-Encoder** 学習に使用する単語ベクトルを Bag-of-Features に固定した場合、正解率は線形分類器では 70.8 % で SdA では 76.9 % であった。単語分散表現を使用した場合、Bag-of-Features は 79.6 % で SdA は 81.7 % であった。このことから、SdA で学習したほうが従来の線形分類器よりも正解率に 2.2 ~ 6.1 ポイントの改善が見

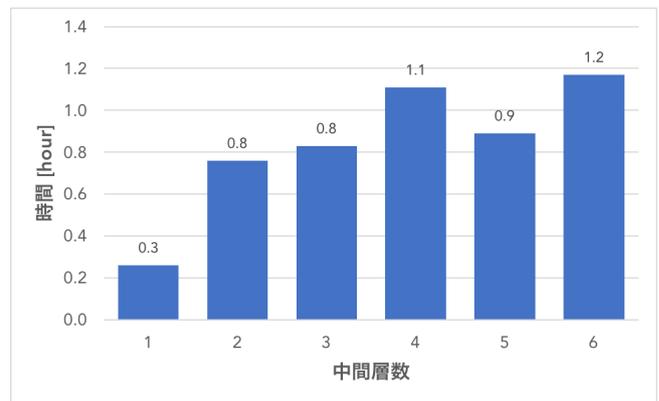


図 3 中間層数の変化に伴う学習時間

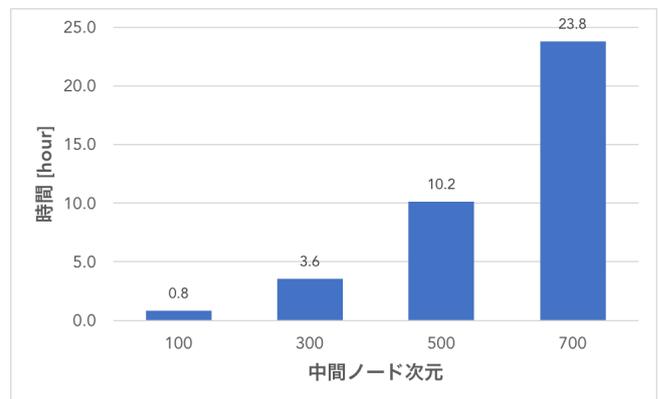


図 4 中間ノード次元の変化に伴う学習時間

られた。したがって、SdA は高い学習性能を発揮したといえる。

**否定表現処理** 表 1 から分かるように、同じパラメータでチューニングしたにもかかわらず、否定表現処理を施した手法はしていない手法と比べて正解率が 0.8 ポイント低下した。これは Nakagawa らの報告とは違うものとなった。考えられる原因として、否定表現処理をすることで引き起こされるスパースネス問題が挙げられる。そこでコーパスに含まれる否定表現の数を調べたところ、異なり語数が 326 語で延べ語数は 1,239 語であった。これはコーパス全体の異なり語数の約 3.8 %、延べ語数の約 1.0 % で正解率に大きな影響を及ぼすと考えにくい。

### 5.2 パラメータ

表 2 にあるように、パラメータを変えることによって得られた結果の考察を行う。

**denoising** ノイズレートが 0 % のまったく denoising を施さない手法 (正解率 81.1 %) と比べると、10 % と少しのノイズを与えたほう (正解率 81.5 %) が正解率が 0.4 ポイントの向上が見られた。しかし過剰にノイズを付与すると正解率が下がることも確認できた。これは強すぎるノイズを与えてしまうと入力を復元で

表 3 事例分析のための正答例と誤答例

正答例						
BoF	Log	AE	Neg	Prop	ans	例文
N	N	N	N	P	P	同 2 5 日の毎日新聞との単独会見では、貧困率などの細かい数字を挙げて 1 0 年間の政権の成果を強調し「フジモリズムはペルー全土に根付いている」と胸を張った。牛で成功したクローン技術の人へ応用するのは難しいことではない。世界経済は、好調を続ける米国とは対照的に、日本や欧州では回復への足取りが重く、日本も今年上半期まで明確な回復軌道が示されなければ、各国が一層の景気回復策を求めてくる可能性がある。
N	P	N	N	P	P	
P	P	P	P	N	N	
誤答例						
BoF	Log	AE	Neg	Prop	ans	例文
N	N	N	N	P	N	もう少し配慮した書き方があったかなとも思う」と反省を口にした。教育省の談話は「歴史教科書は歴史の真相を反映すべきであり、そうしてこそ若い世代に正しい歴史観をもたせ、悲劇の再演を防止できる」と批判した。同市は圧力に屈せず、この宣言を守り抜いてもらいたい。
N	N	N	N	N	P	
P	N	N	P	N	P	

きなくなるという原因が考えられる。

**中間層** 中間層数が 1 のただの AE (正解率 80.6 %) よりも、多層に積み重ねた手法のほう (正解率 81.6 %) が正解率を 1.0 ポイント改善することが見て取れた。これは普通の AE では抽出できないような抽象的な特徴を捉えることができることに起因すると考えられる。

**中間ノード次元** 中間ノードの次元を変化させて実験を行ったが、200 次元の増減で正解率は ± 0.1 ポイントとわずかにしか変動しなかった。顕著な変化を得るためには更に増減幅を増やす必要があると考えられる。

また、パラメータ (中間層数, 中間ノード次元) によっての学習時間の変化をそれぞれ図 3 と図 4 に示す。

図 3 にあるように、中間層の数を増やしていても学習時間の増加は緩やかである。それに対して、中間ノード次元を変化させた場合、図 4 のように 200 次元増やすごとに学習時間は倍になることが見て取れる。これは SdA の構造に起因すると考えられる。SdA において前後の層とのノードの結合は全結合になっている。l 層の n 次元ノードのネットワークを考えると、結合の数は  $l \times n \times n = ln^2$  になる。中間層と結合の数の関係は線形だが、中間ノード次元と結合の数は指数的である。そのため、少しの次元数増加でかなりの時間がかかってしまうと考えられる。また、表 2 から分かるように、中間ノードを 200 次元増やしても正解率は 0.1 ポイントしか変化しないため、中間ノード次元を増やすよりも中間層の数を増加させたほうがより性能の向上につながる。

### 5.3 事例分析

事例分析を表 3 を参考に行っていく。BoF, Log, AE, Neg, Prop, ans はそれぞれ Bag-of-Features, LogRes, Auto-Encoder (1 層の SdA), 否定処理, 提案手法, 正解ラベルに対応する。値には予測値のポジティブとネガティブを指す P と N が表示される。

一番上の正答例に注目すると、「フジモリズム」といった造語に対して、BoF などが苦手としたデータスパースネス問題に提案手法では影響されずに頑健に分類できていることが分かる。また、線形分類器と 1 層の AE では有効な特徴の抽出が困難な中段の二重否定や下段の長い文に対しても、提案手法は上手く対応できていることが分かる。

誤答例の方を見てみると、提案手法は人間的な間違いを犯しているとも見て取れる。例えば上段の例文では「反省」という単語をポジティブと捉えるかネガティブと捉えるかの境界線は曖昧といえる。同じように中段の例文は、「悲劇の再演を防止する」とあるが、最後に「批判した」に続くのでどちらにより重みを置くかは人によって異なると考えられる。実際にこちらのアノテーションタグはネガティブとポジティブ両方付けられていた。下段の例文では「圧力に屈せず」という極性の反転が起きているが、提案手法では上手く対処できなかったが、否定処理を施した手法ではこれに正解した。

## 6. おわりに

本研究では日本語の評価極性分類タスクに対して、word2vec を用いた大規模コーパスからの単語の表現学習と多層 Denoising Auto-Encoder を適用し、高い精度を得ることができた。また、本手法では辞書のような語彙知識や複雑なモデル設計、そして素性エンジニアリングなどの専門家による高度な知識を必要としない。

本研究の展望として、文の分散表現として Recurrent Neural Network [27] や Recursive Neural Network で大域的な情報を獲得して分類を行うことなどが挙げられる。

### 参考文献

- [1] Nakagawa, T., Inui, K. and Kurohashi, S.: Dependency tree-based sentiment classification using CRFs with hidden variables, *Proceedings of Human Language Technologies: The 2010 Annual Conference of the North*

- American Chapter of the Association for Computational Linguistics*, pp. 786–794 (2010).
- [2] Choi, Y. and Cardie, C.: Learning with Compositional Semantics as Structural Inference for Subsentential Sentiment Analysis, *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pp. 793–801 (2008).
- [3] Ikeda, D., Takamura, H., Ratinov, L. and Okumura, M.: Learning to Shift the Polarity of Words for Sentiment Classification, *Proceedings of the 3rd International Joint Conference on Natural Language Processing*, pp. 296–303 (2008).
- [4] Wang, S. and Manning, C. D.: Baselines and bigrams: Simple, good sentiment and topic classification, *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, pp. 90–94 (2012).
- [5] Pang, B., Lillian, L. and Shivakumar, V.: Thumbs up?: sentiment classification using machine learning techniques., *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 79–86 (2002).
- [6] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S. and Dean, J.: Distributed Representations of Words and Phrases and their Compositionality, *Advances in Neural Information Processing Systems 26*, pp. 3111–3119 (2013).
- [7] Mikolov, T., Chen, K., Corrado, G. and Dean, J.: Efficient estimation of word representations in vector space, *International Conference on Learning Representations Workshop* (2013).
- [8] Dahl, G., Dong, Y., Li, D. and Acero, A.: Context-Dependent Pre-Trained Deep Neural Networks for Large-Vocabulary Speech Recognition, *Audio, Speech, and Language Processing, IEEE Transactions*, pp. 30–42 (2011).
- [9] Xie, J., Xu, L. and Chen, E.: Image Denoising and Inpainting with Deep Neural Networks, *Advances in Neural Information Processing Systems 25*, pp. 341–349 (2012).
- [10] Wang, N. and Yeung, D.-Y.: Learning a Deep Compact Image Representation for Visual Tracking, *Advances in Neural Information Processing Systems 26*, pp. 809–817 (2013).
- [11] Chen, M., Xu, Z., Weinberger, K. and Sha, F.: Marginalized denoising autoencoders for domain adaptation, *Proceedings of The 29th International Conference on Machine Learning* (2012).
- [12] McCallum, A.: Efficiently inducing features of conditional random fields, *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pp. 403–410 (2002).
- [13] Collobert, R. and Weston, J.: A unified architecture for natural language processing: Deep neural networks with multitask learning, *Proceedings of the 25th International Conference on Machine Learning*, pp. 160–167 (2008).
- [14] Glorot, X., Bordes, A. and Bengio, Y.: Domain adaptation for large-scale sentiment classification: A deep learning approach, *Proceedings of the 28th International Conference on Machine Learning*, pp. 513–520 (2011).
- [15] dos Santos, C. N. and Gatti, M.: Deep convolutional neural networks for sentiment analysis of short texts, *Proceedings of the 25th International Conference on Computational Linguistics*, pp. 69–78 (2014).
- [16] Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T. and Qin, B.: Learning sentiment-specific word embedding for twitter sentiment classification, *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, Vol. 1, pp. 1555–1565 (2014).
- [17] Socher, R., Pennington, J., Huang, E. H., Ng, A. Y. and Manning, C. D.: Semi-Supervised Recursive Autoencoders for Predicting Sentiment Distributions, *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing* (2011).
- [18] Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y. and Potts, C.: Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank, *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631–1642 (2013).
- [19] Bengio, Y., Ducharme, R., Vincent, P. and Janvin, C.: A neural probabilistic language model, *The Journal of Machine Learning Research*, Vol. 3, pp. 1137–1155 (2003).
- [20] Le, Q. and Mikolov, T.: Distributed Representations of Sentences and Documents, *Proceedings of the 31st International Conference on Machine Learning*, pp. 1188–1196 (2014).
- [21] Hinton, G. E. and Salakhutdinov, R. R.: Reducing the dimensionality of data with neural networks, *Science*, Vol. 313, No. 5786, pp. 504–507 (2006).
- [22] Vincent, P., Larochelle, H., Bengio, Y. and Manzagol, P.-A.: Extracting and Composing Robust Features with Denoising Autoencoders, *Proceedings of the 25th International Conference on Machine Learning*, pp. 1096–1103 (2008).
- [23] Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R. R.: Improving neural networks by preventing co-adaptation of feature detectors, *arXiv preprint arXiv:1207.0580* (2012).
- [24] Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y. and Manzagol, P.-A.: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion, *The Journal of Machine Learning Research*, Vol. 11, pp. 3371–3408 (2010).
- [25] Seki, Y., Evans, D. K., Ku, L.-W., Chen, H.-H., Kando, N. and Lin, C.-Y.: Overview of opinion analysis pilot task at NTCIR-6, *Proceedings of NTCIR-6 Workshop Meeting*, pp. 265–278 (2007).
- [26] 中川哲治, 乾健太郎, 黒橋禎夫: 隠れ変数を持つ条件付き確率場による依存構造木の評価極性分類, 研究報告自然言語処理 (NL), Vol. 2009, No. 10, pp. 1–7 (2009).
- [27] Irsoy, O. and Cardie, C.: Opinion mining with deep recurrent neural networks, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 720–728 (2014).