Word2Vec と自己組織化マップを用いた文書分類

吉岡宏樹[†1]　堂園浩[†1]

概要：SNS の普及，Web ページの増加により，テキストデータを扱う機会が増えている．膨大なテキストデータを手作業で処理するのは困難であるため．現在様々な機械的な分類手法が提案されている．本論文では Word2Vec と自己組織化マップを用いた新たな分類手法を提案し，その能力を実験により確認した．

キーワード：Word2Vec，自己組織化マップ，文書分類

The Classification of the Documents Based on Word2Vec and 2-layer Self Organizing Maps

KOKI YOSHIOKA [†1]　HIROSHI DOZONO[†1]

Abstract: Due to popularization of SNS and increase of web pages, people have to deal with more text data. However, it is difficult to process huge text data manually. Therefore, various mechanical classification methods have been proposed. Also in this paper I would like to propose a classification method using Word2Vec and SOM, and examine the performance with experiments.

Keywords: Word2Vec, Self Organizing Map(SOM), Documents classification

1. はじめに

SNS の普及，Web ページの増加により様々なテキストデータを扱う機会が増えている．日々増えていく膨大なテキストデータに対して，人手で処理を行うのは困難である．今後テキストデータはさらに増え続けていくと予想されるため，テキストを機械に分類させる技術がさらに必要となる．

テキスト分類を機械に行わせる場合，テキストを機械的に扱えるデータ形式に変換しなければならない，代表的な手法の 1 つとして BoW モデル[1] により特徴データに変換する方法がある．しかし，BoW モデルにより作成した特徴データでは，1 度でも出てきた単語に対しても，1 方向のベクトルを割り当てるため，分類対象の文書が多くなると次元数が膨大となる．そこで通常なんらかの方法で次元圧縮をする．本稿では自己組織化マップ[2]と近年，自然言語処理の分野で多く使用されている Word2Vec[3] を用いた新たな BoW モデルを改良した，テキストデータの変換方法を提案するとともに，そのデータを再び自己組織化マップに学習させてテキストデータの分類結果を確認する．分類対象の文書は Livedoor ニュースコーパス[4]を使用した．

2. 関連知識

2.1 Bag of Words モデル (BoW)

単語の出現頻度によって文書や文章を分類する場合に用いられる手法である．分類する文書に，出現する単語をカウントして各文書の要素とすることで文書をベクトルで表す． BoW モデルは単語もベクトルで表すことができ，表したい単語の番号の要素を 1 としてそれ以外を 0 とすることで表すことができる．例として表 1 に文書が 3 種類存在する場合を示す．これは文書 1 には単語 1 が 3 語存在し，文書 2 には単語 1 が 2 語存在することを表している．文書全体の単語は n 種類存在していて，文書の数が増えれば出現する単語の種類も増えるため次元数も増えていく．n 次元のベクトルの最初の要素だけを 1 として他の要素を 0 とすることで単語 1 をベクトルで表せる．

表 1 BoW モデル

	単語1	単語2	単語3	単語4	単語5	…	単語n
文書1	3	2	0	1	0		5
文書2	2	1	1	0	3		2
文書3	3	0	0	0	2		1

2.2 自己組織化マップ(Self-Organizing Map : SOM)

自己組織化マップは T・コホネンによって提案されたニューラルネットワークの 1 つである．教師なし学習によって高次元データを低次元マップへと写像し，データ間の類似度をマップ上での距離によって表現する．類似度が高いデータ同士ほど距離は近くなり，低ければ遠くなるためデータの可視化に利用される．[2]　基本的に図 1 に示すように SOM は入力層と競合層の 2 層のニューラルネットワー

†1 佐賀大学大学院工学系研究科先端融合工学専攻
Department of Advanced Technology Fusion, Graduate School of Science and Engineering Saga University

クである．

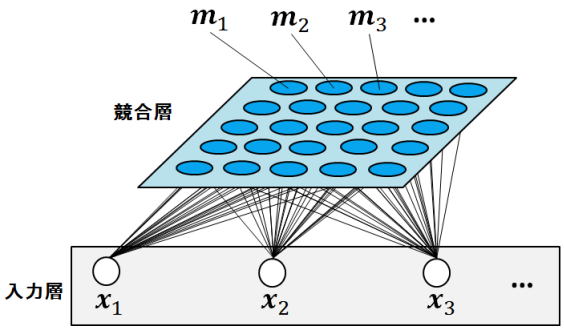

図 1　SOM の構造

2.2.1 SOM のアルゴリズム

SOM は一般的に 2 次元格子状に配置されたノードにより構成される．各ノードは乱数により初期化された参照ベクトル$\boldsymbol{m_i}(m_{i1}, m_{i2}, \cdots)$を持ち，入力ベクトルの集合から選ばれた$\boldsymbol{x_j}(x_{j1}, x_{j2}, \cdots)$に，最もユークリッド距離が近い参照ベクトルを持つノードを，最整合ノードC_jとし，各ノードが持つ参照ベクトルを近傍関数$h_{ci}(t)$により更新する．入力ベクトルの選択から参照ベクトルの更新を繰り返すことで SOM の学習が進む．以下に更新の式を示す．

$$\boldsymbol{m}_i(t+1) = \boldsymbol{m}_i(t) + h_{ci}\left[\boldsymbol{x}_j(t) - \boldsymbol{m}_i(t)\right] \tag{1}$$

$$h_{ci}(t) = \alpha(t)\exp\left(-\frac{\|\boldsymbol{r}_c - \boldsymbol{r}_i\|^2}{2\sigma^2(t)}\right) \tag{2}$$

$\alpha(t)$は学習率係数，$\sigma(t)$は近傍半径である．$\alpha(t)$，$\sigma(t)$ は学習が進むにつれて減少するように設定する．$\boldsymbol{r}_c$，$\boldsymbol{r}_i$ はマップ上での最整合ノードと更新するノードの位置ベクトルである．

2.2.2 バッチ学習 SOM

2.2.1 で示した学習方法は，オンライン学習と呼ばれ，一般的な SOM の学習方法である．しかし，大量の入力ベクトルを学習させる場合，学習させる順番によって結果が異なってくる．本実験でも数千から数万の入力ベクトルを扱うため，この影響は無視できない．そこで，結果が学習順番に依存しないバッチ学習[4]で実験を行う．バッチ学習は先にすべての最整合ノードを見つけて，次の式で参照ベクトルを更新していく．Nは入力ベクトルの数である．

$$\boldsymbol{m}_i(t+1) = \frac{\sum_{j=1}^{N} h_{cj,i}\,\boldsymbol{x}_j}{\sum_{j=1}^{N} h_{cj,i}} \tag{3}$$

$$h_{cj,i}(t) = \exp\left(-\frac{\|\boldsymbol{r}_c - \boldsymbol{r}_i\|^2}{2\sigma^2(t)}\right) \tag{4}$$

2.2.3 球面 SOM

通常 SOM は 2 次元平面にマッピングされるが，球面上で学習させることで，マップの端をなくした球面 SOM[6]が提案されている．本稿でも球面 SOM を用いて実験を行った．ノードは正二十面体を基礎とした，ジオデシックドームの頂点に配置する手法を用いた．この場合n回分割したノード数は

$$2 + 10 \times 4^n \tag{5}$$

となる．

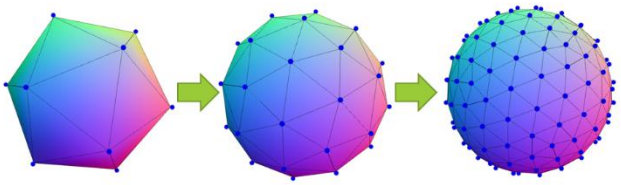

図 2 球面 SOM のノード配置

2.3 WEB-SOM

WEB-SOM [7]は，文書の BoW モデルを入力ベクトルとして WEB データのような大量の文書を分類する自己組織化マップである．入力ベクトルの次元を減少させるために，単語は一度，単語カテゴリマップによって分類され，単語カテゴリマップ上にマップされた文書は，文書マップ上に分類される．WEB-SOM は、同様の WEB ページを検索できる検索エンジンとしても利用できる．本稿では，Word2Vec を単語カテゴリマップの前処理に導入し，単語の意味をより効率的に考慮した単語ベクトルの次元を圧縮する．

2.4 Word2Vec

文章中の単語を任意の次元ベクトルに変換するツールである．単語をベクトルに変換することによって単語と単語の類似度の確認や， 単語同士の線形計算を行うことができる．

3. 実験方法

3.1 Word2Vec による単語のベクトル化

分類対象とする文書をすべて Word2Vec に読み込ませ出現するすべての単語にベクトルを割り当てる．Word2Vec にテキストを読み込ませるには，単語の間をスペースで分ける必要がある．本実験では日本語のテキストを扱うため，英語と違い，分かち書きを行う．分かち書きには形態素解析ソフトの MeCab を使用した．MeCab の辞書には ipadic-NEologd を使用した．

3.2 単語を SOM に学習させる

単語に Word2Vec でベクトルを割り当てたら，そのベク

トルを入力ベクトルとして SOM に学習させる．通常，SOM に学習させる場合，ノードの数を入力ベクトルよりも多く設定するが，ここではあえて入力ベクトルよりも，ノードの数を減らす．これは SOM の性質を利用して類似性の高い単語を同じ座標のノードへと格納するためである．以降単語を学習したマップを単語マップと呼ぶ．

3.3 各文書の入力ベクトルの作成

単語を SOM で学習した後，各文書の入力ベクトルを作成する．各文書に単語マップと同じノード数のマップを割り当て，それぞれの文書に存在する各単語の数を，単語マップにおける最整合ノードと同じ座標に格納する．図 3，図 4 に例を示す．文書 1 には単語 11 と単語 13 が合計で 2 語含まれていることを意味する．座標が重なった単語を，同じ単語としてカウントすることで次元を圧縮する．最後に，各文書のマップを入力ベクトルとして，SOM に学習させて結果を確認する．図では説明のために 2 次元平面でマップを作成しているが，実験では球面 SOM を用いて実験を行う．

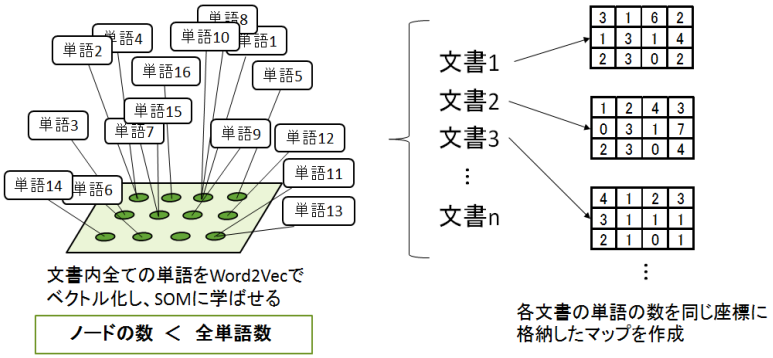

図 3 各文書の入力データの作成

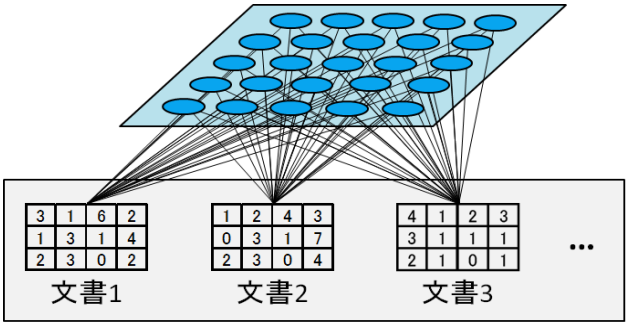

図 4 SOM で文書マップ分類

4. 実験結果

4.1 全種類の単語の場合

Livedoor ニュースコーパスは 7367 件の記事があり，9 カテゴリに分けられている．これらを Word2Vec に読み込ませ，単語に 100 次元のベクトルを割り当てた．文章中に 5 回以上出現しない単語については，破棄するように設定していたため，ベクトルを割り当てられた単語は，助詞や記号を含め 34772 種類だった．単語マップのノードを 162 として，各文書の入力ベクトルを作成し，ノード数 10242 の SOM に学習させた．結果を図 5 に示す．結果はカテゴリごとに色分けしてプロットしている．最整合ノードが同じ場合は，球面垂直方向に重ねてプロットしている．左と右で見る角度を変えて表示している．結果を見ると，差はあるが，同じカテゴリ同士でグループを形成していることがわかる．

4.2 特定の単語の場合

Livedoor ニュースコーパスから，名詞，動詞，形容詞以外を取り除いて，分かち書きし，Word2Vec に読み込ませた．ベクトルを割り当てられた単語の種類は 33547 種類となった．単語マップのノード数を 162 として，同じようにノード数 10242 の SOM に学習させた．結果を図 6 に示す．全単語の場合と比較しても結果に大きな違いは見られなかった．

5. まとめ

本稿では Word2Vec と SOM を用いて，類似度が高い単語を同じ単語としてカウントし，BoW モデルよりも低次元な特徴データを作成する方法を提案した．結果は各カテゴリのデータごとに何らかの特徴を示した．確認された特徴は，各文書内に存在している単語に依存しているものだと推測できるため，目的である BoW モデルの特徴を保持しながらのデータ量の削減はできたと考えられる．今回の実験では結果の評価をカテゴリで行っていたため，今後は同じカテゴリ同士のデータの関係性や，単語マップのノード数による結果の影響を見ていきたい．

6. 参考文献

[1] Bag of Words (BoW) - Natural Language Processing, *https://ongspxm.github.io/blog/2014/12/bag-of-words-natural-language-processing/*

[2] T. ~Kohonen, Self Organizing Maps, Springer, ISBN 3-540-67921-9, 2001

[3] Mikolov, Tomas; Sutskever, Ilya; Chen, Kai; Corrado, Greg S.; Dean, Jeff (2013). Distributed representations of words and phrases and their compositionality. Advances in Neural Information Processing Systems

[4] Download: Livedoor new corpus, https://www.rondhuit.com/download.html

[5] Haruna Matsushita and Yoshifumi Nishio (2010) “Batch-Learning Self-Organizing Map with Weighted Connections Avoiding False-Neighbor Effects” http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.700.1701&rep=rep1&type=pdf

[6] 徳高平蔵，大北正昭，藤村喜久郎，(2007)，『自己組織化マップとその応用』シュプリンガー・ジャパン

[7] Samuel Kaski, Timo Honkela, Krista Lagus, Teuvo Kohonen, WEBSOM – Self-organizing maps of document collections, Neurocomputing, Volume 21, Issues 1–3, 6 November 1998, Pages 101-117

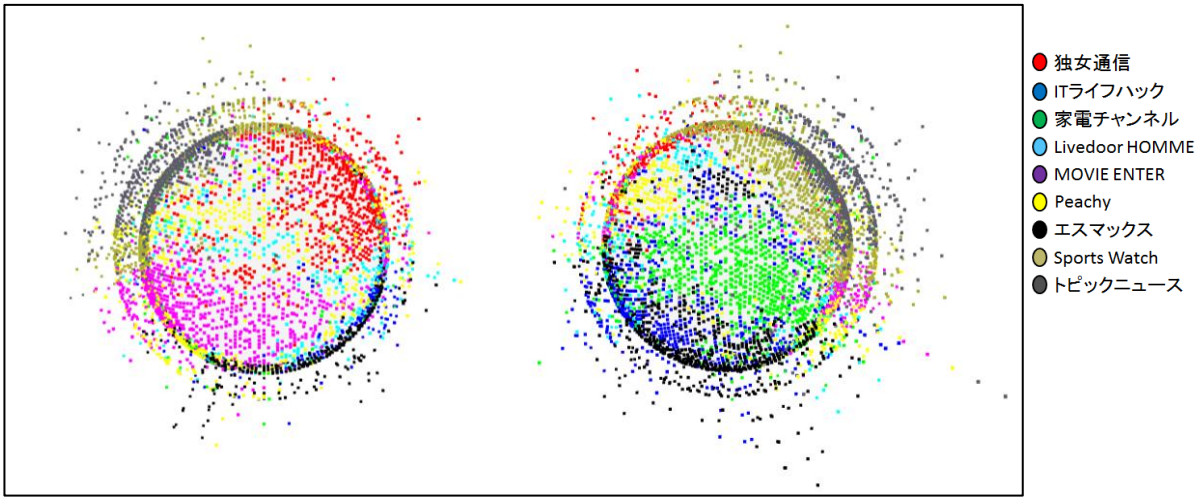

図 5 全単語の結果

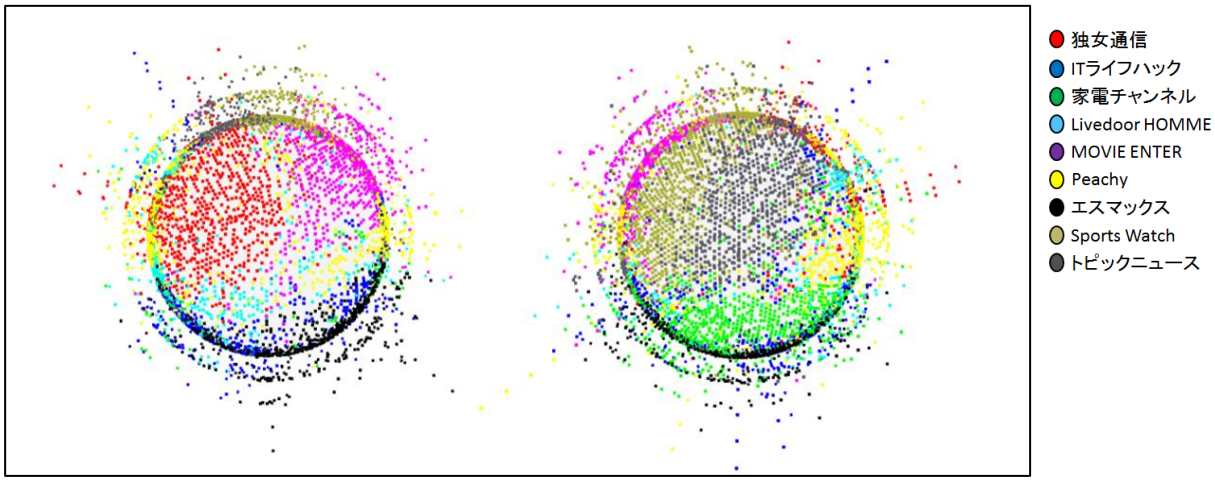

図 6 特定単語の結果