The 32nd Annual Conference of the Japanese Society for Artificial Intelligence, 2018

Word2Vecにおける出力側の重みに注目した文書分類手法の検討

A study on document classification focusing on the output side weight on Word2Vec

内田脩斗 *1 Shuto Uchida　吉川大弘 *1 Tomohiro Yoshikawa　ジメネスフェリックス *1 Felix Jimenez　古橋武 *1 Takeshi Furuhashi

*1名古屋大学大学院工学研究科
Graduate School of Engineering Nagoya University

Document classification is an important technology in modern information society. In recent years, distributed representation (DR) which embeds semantic relationships of words into vectors has attracted attention and the methods applying DR to document classification have been reported. DR can be generated mainly by using a tool called Word2Vec. Word2Vec has the learning structure using a neural network, and we use the weights on the input side as DR. However, Word2Vec learns different characteristic weights on the output side from DR, which is not focused on and not commonly used. In this paper, we propose a document classification method by ensemble learning using DR and the output side weights and suggest the usefulness on the proposed method.

1. はじめに

インターネットの普及に伴って大量の電子文書が日々生成されている現代において，テキストデータの自動解析手法や解析技術は，様々な場面での応用が期待されている．その技術の一つに文書分類がある．文書分類とは，与えられた文書を予め定められたクラスのいずれかに分類することである．スパムメール分類や Web 記事分類などに広く実用化されており，より高精度な分類手法が求められている．

文書分類では各文書をどのような素性で表すかが重要であり，各単語は互いに独立であるという仮定の下，単語の出現回数を利用する Bag-of-Words（BOW）という表現手法が古くから広く利用されている [1][2][3].

一方で，Micolov らが発表した Word2Vec[4] は，大規模コーパスから教師なし学習を行うことで，語義の似た単語が類似した重みの要素をもつ分散表現を生成することができる．これにより，従来では困難であった単語ベクトル間での意味の演算が可能となり，学習された分散表現に対し，

$$vector(Paris) - vector(France) + vector(Italy)$$

により算出されるベクトルが $vector(Rome)$ に，

$$vector(king) - vector(man) + vector(woman)$$

により算出されるベクトルが $vector(queen)$ に，それぞれ近くなるという性質をもっている．このように分散表現は，学習によって得られた単語同士の意味的な関係を捉えることができ，文書分類に適用する手法も多く報告されている [5][6].

この Word2Vec は，ニューラルネットワークを利用した学習構造をしており，ネットワーク上の入力層-隠れ層間の重みである上述の分散表現と，それと共に生成される隠れ層-出力層間の重み，“出力側の重み”があるが，一般的に利用されることはない．しかし，この出力側の重みは単語に対して分散表現とは異なる意味関係を捉えていると考えられ，その有用性があると思われる．

連絡先: 内田脩斗，名古屋大学大学院工学研究科，名古屋市千種区不老町，052-789-2793，052-789-3166，uchida@cmplx.cse.nagoya-u.ac.jp

そこで本稿では，分散表現と出力側の重みの捉えている意味関係の違いについて検討し，分散表現と出力側の重みをアンサンブル学習により組み合わせた文書分類手法を提案する．

2. Word2Vec

本章では．Word2Vec の学習モデルについて説明する．

Word2Vec は，言語処理でよく用いられる分布仮説（同じ文脈で出現する単語は同じ意味を持つこと）[7] に基づいており，文脈上のある単語に対して，周辺に現れやすい単語を予測することをモデル化した構造をしている．

図 1 は Word2Vec の学習モデルを表した図である．また，入力層と出力層の次元数はボキャブラリ数，隠れ層は埋め込み次元数である．Word2Vec はニューラルネットワークを用いて

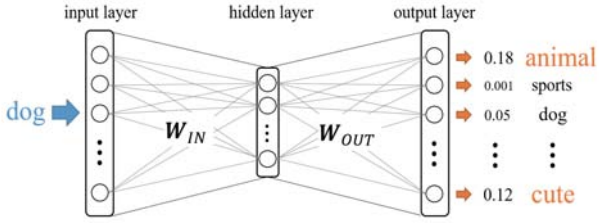


図 1: Word2Vec 学習モデル図

分散表現を獲得する．具体的には，入力層に入力された単語に対して，その単語の周辺に出現しやすい単語の出現確率が大きくなるように各層の重み（入力側重み W_{IN}，出力側重み W_{OUT}）を更新する．例えば，ある文脈上において,「dog」という単語の周辺に「animal」や「cute」という単語が出現したとすると，入力層の「dog」に対応した要素に 1，それ以外の要素に 0 が入力され,「animal」,「cute」に対応した出力層の出力が 1 に近づくように W_{IN} と W_{OUT} が更新される．

W_{IN} は周辺単語の情報を元に，単語の意味関係を学習している．つまり,「dog」や「cat」という単語の周辺に「animal」や「cute」という単語が共通して出現することで類似した重みを学習し，

$$vector(dog) \simeq vector(cat) \simeq vector(rabbit)$$

に近づくように，個々の次元に単語の意味関係を埋め込んでいる．また，W_{IN} の各行ベクトルが個々の単語ベクトルに対応しており，Word2Vec では W_{IN} を分散表現として利用することを前提とした学習モデルとなっている．

一方，W_{OUT} はある単語の周辺に出現する単語を予測するための重みである．つまり，「dog」や「cat」という単語の周辺に「animal」や「cute」という単語が現れる確率を大きくするため，入力単語のベクトルと周辺単語のベクトルの内積値が大きくなるように重みを学習する．これより，W_{OUT} は単語ベクトルを共起しやすい単語に展開する共起単語ベクトルであると捉えることができる．つまり，W_{OUT} においては，

$$vector(animal) \simeq vector(cute) \simeq vector(bark)$$

のような共起関係を学習する傾向にある．また，W_{OUT} の各列ベクトルが個々の共起単語ベクトルに対応している．ただし，W_{OUT} は W_{IN} を獲得するために生成される副産物と考えられており，一般的に W_{OUT} を単語ベクトルとして利用することはない．W_{OUT} を利用した従来研究は数少ないが，Mitra ら [8] は，W_{IN} と W_{OUT} の双方から得られる共起情報を利用した文書検索手法を提案している．しかし，W_{OUT} を単語ベクトルとして，独立して用いる本研究とは異なっている．

上述したように，W_{IN} と W_{OUT} では学習される重みの性質が異なっており，通常利用されない W_{OUT} にも有用性があると考えられる．

3. 従来手法

Word2Vec により学習された分散表現 W_{IN} を，文書分類に適用する一般的な方法 [5] を説明する．ある文書 d_i があったとき，その文書のベクトル表現である文書ベクトル $\vec{d_i}$ は，以下の式で算出される．

$$\vec{d_i} = \frac{1}{|d_i|} \sum_{j=1}^{|d_i|} \vec{w_j} \quad (1)$$

$\vec{w_j}$ は文書 d_i における j 番目の単語の単語ベクトルである．取得する単語ベクトルが分散表現に含まれていない場合，その単語は（そもそも文書内に含まれていなかったものとして）除外する．これは，分散表現を学習したコーパスに存在しない単語が文書 d_i に含まれていた場合に生じる．$|d_i|$ は，文書内の単語のうち，単語ベクトルを取得できた単語の数である．

(1) 式は，文書内に含まれる単語の単語ベクトルの重心（平均）をとったものを文書ベクトルとして扱うことを意味する．この文書ベクトルを用いて分類を行う．

従来手法では，W_{IN} のみを文書分類に適用する．一方，W_{OUT} は W_{IN} と異なる情報を保持しており，これを利用することで更なる分類精度の向上に寄与できると考えられる．そこで本稿では，W_{IN} と W_{OUT} を組み合わせた分類手法を提案する．

4. 提案手法

4.1 アンサンブル学習

提案手法では，[9] で利用されているクラス所属確率を用いたアンサンブル学習を行う．アンサンブル学習とは，複数の分類器を組み合わせることで分類精度を向上させる手法である．また，クラス所属確率とは，分類対象となる文書がそれぞれのクラスに所属する確率である．

4.2 概要

提案手法のイメージを図 2 に示す．

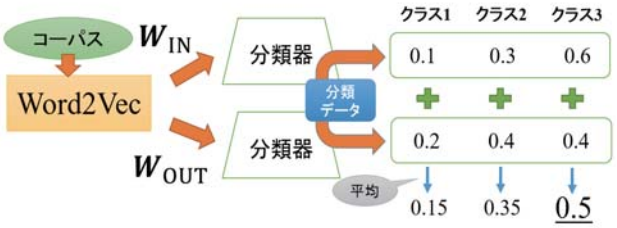


図 2: 提案手法のイメージ

1. テキストコーパスから Word2Vec を用いて，W_{IN} と W_{OUT} を生成する．(W_{IN} と W_{OUT} は同時に生成される)
2. 個々の素性を用い，文書ベクトルで表した学習文書に対してそれぞれ分類器を生成する．
3. 分類データに対して個々の分類器によりクラス所属確率を算出する．
4. クラスごとの所属確率の平均値を算出し，最大のクラスに文書を分類する．

5. 実験設定

5.1 分類対象データセット

実験で使用した分類対象データセットを表 1 に示す．livedoor ニュースコーパス（以下、「livedoor」という）は日本語のテキストデータとなっており，ウェブサイト *1 からダウンロードして利用できる．また，Reuters 21578 は英語のテキストデータとなっており，[10] の著者のウェブサイト *2 からダウンロードして利用できる．

表 1: 分類対象データセット

データセット	学習文書数	テスト文書数	クラス数
livedoor	4420	2947	9
Reuters 21578	5485	2189	8

5.2 Word2Vec 学習コーパス

Word2Vec の学習には日本語 Wikipedia と英語 Wikipedia を利用した．また，各種パラメータは，$window = 5$，$size = 300$ を使用した．「$window$」は前後何単語を教師データとするかを指定するオプション，「$size$」は学習する単語ベクトルの次元数を指定するオプションである．Word2Vec の実装は Python の gensim ライブラリを使用し，また，W_{OUT} はライブラリ内の syn1neg に保存されているものを使用した．

5.3 分類器

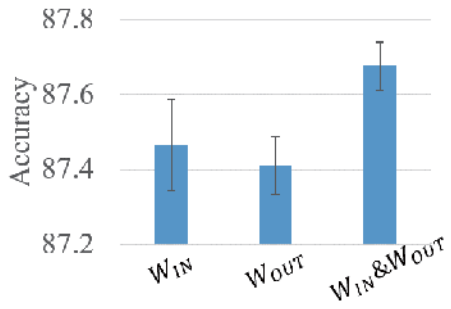
分類器には SVM を用いた．SVM は RBF カーネルを用い，ハイパーパラメータはライブラリのデフォルト値である $C = 1$ と $\gamma = 1/$次元数 (300) を用いた．

*1 https://www.rondhuit.com/download.html
*2 http://web.ist.utl.pt/acardoso/datasets/

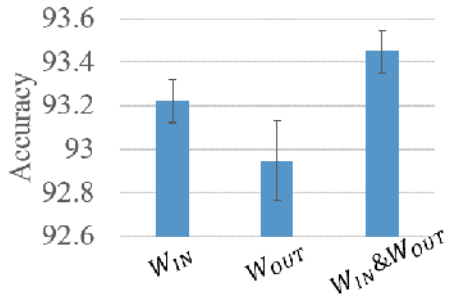
6. 実験

6.1 実験 1（提案手法の有効性の検証）

実験 1 では，従来手法に対する提案手法の有効性の検証を行った．図 3(a) は日本語 Wikipedia から学習した Word2Vec における W_{IN} と W_{OUT} を用いて，livedoor を分類した際の正答率，図 3(b) は英語 Wikipedia から学習した Word2Vec における W_{IN} と W_{OUT} を用いて Reuters 21578 を分類した際の正答率である．また，Word2Vec は内部にランダム性を保持しているため，ここでは 5 試行の平均と標準誤差を表示している．



(a) livedoor ニュースコーパス



(b) Reuters 21578

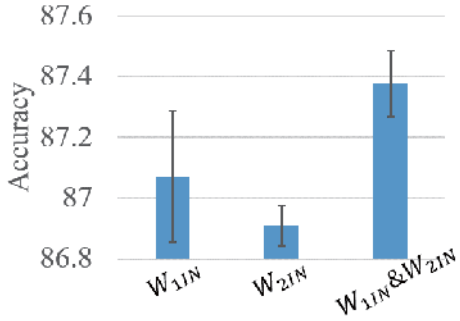
図 3: 各データセットにおける正答率

正答率は，テスト文書に対して，正しく分類された割合である．また，図の W_{IN} は，分類器の素性として W_{IN} を用いた場合の正答率（従来手法），W_{OUT} は，分類器の素性として W_{OUT} を用いた場合の正答率（参考），$W_{IN}\&W_{OUT}$ はアンサンブル学習を用いた場合の正答率（提案手法）を示している．

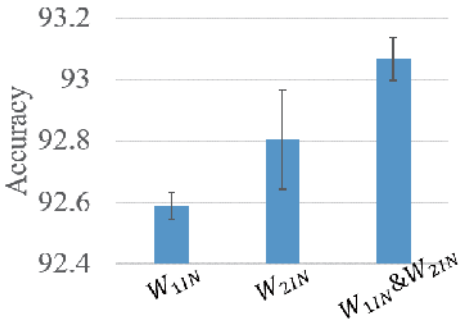
これらの結果より，W_{OUT} 単体では従来手法の W_{IN} に劣っているが，W_{IN} と W_{OUT} を組み合わせた提案手法では精度の向上が確認でき，提案手法が有効である可能性を確認した．これは，W_{IN} と W_{OUT} が異なる特徴を捉えていることで，個々の分類器に強みがあり，アンサンブル学習が有効に働いているものと考えられる．

6.2 実験 2（他手法との比較）

実験 2 では，同一のコーパスから異なる分散表現を生成する手法を用いて，アンサンブル学習を行った場合との比較を行った．図 4(a) に livedoor を分類した際の正答率（W_{1IN} と W_{2IN}），図 4(b) に Reuters 21578 を分類した際の正答率（W_{1IN} と W_{2IN}）を示す．実験 1 と同様に，5 試行での平均と標準誤差を表示している．



(a) livedoor ニュースコーパス



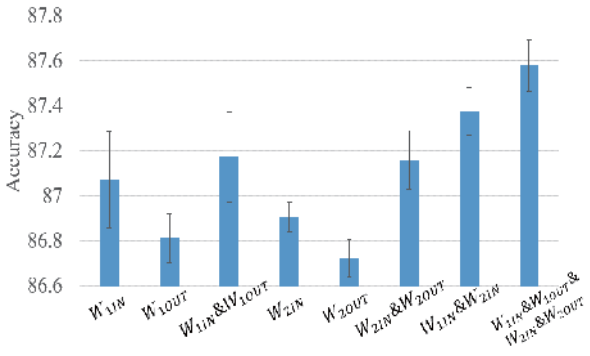
(b) Reuters 21578

図 4: 各データセットにおける正答率（W_{1IN} と W_{2IN}）

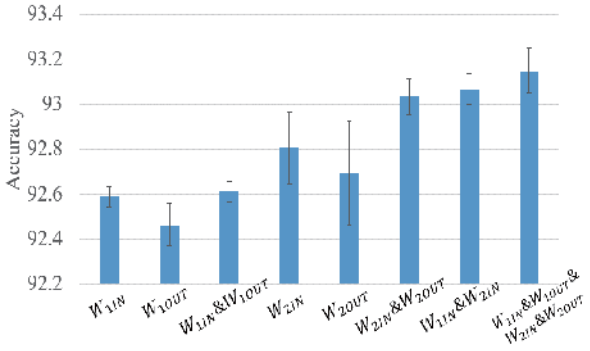
同一のコーパスから異なる分散表現を獲得するため，Wikipedia コーパスを 2 分割し，それぞれに対して Word2Vec により学習した 2 種類の W_{IN} である W_{1IN} と W_{2IN} を生成した．コーパスを分割することで出現単語や文書情報に偏りが生じるため，生成される W_{1IN} と W_{2IN} の特徴の捉え方に差が生じ，双方を用いてアンサンブル学習を行った場合にも，分類精度の向上が期待できる．実際に図 4(a) と図 4(b) より，精度の向上が見て取れるが，実験 1 の従来手法 W_{IN} よりも精度が劣っていることが確認できる．これは，コーパスを分割したことで Word2Vec の学習に使用するデータ量が減少したことによる影響であると考えられる．よって，コーパスのサイズを維持しつつ，異なる意味関係を捉える W_{IN} と W_{OUT} を使用する有効性が確認された．

さらに，提案手法を実験 2 に適用した場合の結果を図 5(a) と図 5(b) に示す．

これより，W_1 と W_2 のそれぞれで，実験 1 と同様の傾向が確認でき，提案手法の有効性が確認された．これらの結果から，複数のコーパスを Word2Vec の学習に利用し，提案手法と組み合わせることで，更なる精度の向上に繋がることが期待できる．W_{OUT} は，W_{IN} を生成するときに同時に生成される



(a) livedoor ニュースコーパス



(b) Reuters 21578

図 5: 各データセットにおける正答率（W_1 と W_2）

ため，分散表現を用いた文書分類手法には常に適用することが可能と考えられ，一般的に利用されてこなかった W_{OUT} の有用性が示唆された．

7. まとめ

本稿では，Word2Vec の学習過程で生成される出力側の重み W_{OUT} に注目し，入力側の重み W_{IN} と出力側の重み W_{OUT} を利用したアンサンブル学習による文書分類手法を提案した．実験の結果，提案手法の有効性を確認した．また，一般的には用いられていない出力側の重み W_{OUT} の新たな有用性を示した．

今後は，他のデータセットに対する有効性の検証や，W_{OUT} のより有効的な活用方法について検討することが挙げられる．

参考文献

[1] Pedro Domingos and Michael J. Pazzani. ”On the optimality of the simple Bayesian classifier under zero-one loss.” Machine Learning, 29(2-3):103130, 1997.

[2] Rennie J., Shih, L., Teevan, J., Karger, D. ”Tackling the Poor Assumptions of Naive Bayes Text Classifiers.” Proc. of ICLM-2003.

[3] Rie Johnson and Tong Zhang. ”Effective use of word order for text categorization with convolutional neural networks.” arXiv preprint arXiv:1412.1058, 2014.

[4] Mikolov, Tomas, et al. ”Efficient estimation of word representations in vector space.” arXiv preprint arXiv:1301.3781 (2013).

[5] Xing, Chao, et al. ”Document classification with distributions of word vectors.” Asia-Pacific Signal and Information Processing Association, 2014 Annual Summit and Conference (APSIPA). IEEE, 2014.

[6] R. Liu, D. Wang, and C. Xing, ”Document classification based on word vectors.” in ISCSLP ’ 14, 2014.

[7] Hinton, G.E., McClelland, J.L., Rumelhart, D.E. ”Distributed representations.” In: McClelland, J.L., Rumelhart, D.E. (Eds.), Parallel Distributed Processing: Explorations in the Microstructure of Cognition. MIT Press, Cambridge, MA, pp. 77109, 1986.

[8] Bhaskar Mitra, Eric Nalisnick, Nick Craswell, Rich Caruana. ”A dual embedding space model for document ranking.” arXiv preprint arXiv:1602.01137. 2016.

[9] 高橋和子. ”多クラス SVM におけるクラス所属確率を用いたアンサンブル学習の提案.” 研究報告音声言語情報処理 (SLP) 2011.2 (2011): 1-8.

[10] Cardoso-Cachopo, A. ”Improving Methods for Single-label Text Categorization.” PdD Thesis, Instituto Superior Tecnico, Universidade Tecnica de Lisboa, 2007.