1K3-3

複数のコーパスから学習した分散表現の利用による文書分類精度向上に関する検討

A study on improving accuracy of document classification using sets of word vectors learned from corpora

宅和晃志 *1 Koji Takuwa　吉川大弘 *1 Tomohiro Yoshikawa　古橋武 *1 Takeshi Furuhashi

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

Document Classification (DC) is an important task in digital society. In DC, it is important how to represent a document. Typical DC methods represent a document as Bag-of-Words (BOW) which uses only the number of occurrences of each word, which ignores the semantic meaning of words. Recent years, DC methods using Word2Vec are proposed and got much attention. Word2Vec is a tool for learning semantic-syntactic relationships among words as word vectors. The DC methods using Word2Vec represent a document as the centroid of word vectors in a document and use only semantic meaning of each word, which ignores the number of occurrences of words. In this paper, we propose a new DC method combining BOW and some sets of word vectors. Occurred words and wording will be different between corpora, so each set of word vectors learned from each corpus is expected to represent different semantic meaning.

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．そのうち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．

文書分類において，NaiveBayes 分類器 [1] はその処理の速さから幅広く用いられている手法であり，SVM[2] はその分類精度の高さが広く知られている．従来の文書分類では一般的に，各単語が互いに独立だと仮定して各単語の出現回数のみを利用する Bag-of-Words（BOW）という表現手法が用いられてきた．しかし，BOW では単語の頻度情報しか利用しないため，同義語や表記揺れなどの意味的に近い単語を，全く別の単語として扱ってしまうという問題点がある．

一方で，近年 word2vec[3] の登場により，文書分類に分散表現を適用する手法が数多く報告されている [4][5][6][7]．word2vec は，大規模コーパスから教師なしで単語の分散表現を学習する手法である．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，それぞれ近くなる性質を持っている．このように，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る．分散表現の文書分類への適用としては，文書内の単語ベクトルの平均を文書ベクトルとして用い，SVM や k 近傍法などの分類手法により分類する方法が一般的である．しかし，この方法では，最終的に単語ベクトルの平均をとってしまうため，BOW では捉えることができていた各単語の頻度情報が失われてしまうという問題点がある．

本稿では，頻度情報を捉えることができる BOW と，単語間の意味的な関係を捉えることができる分散表現を組み合わせる方法について検討する．さらに，分散表現に対しては，単一の分散表現に限らず，複数のコーパスからそれぞれ学習された複数の分散表現を利用する．単一のコーパスを用いた場合，学習される分散表現に意味的な偏りが生じると考えられる．例えばコーパスとして Wikipedia を用いた場合，辞書的な単語や言い回しが多いことで，学習された分散表現において，Wikipedia に偏った意味関係が学習されてしまうと考えられる．それに対して，複数種類のコーパスを用いた場合，それぞれのコーパスで使われている単語やその使われ方（共起の仕方）が異なることから，学習される分散表現はそれぞれ異なった意味関係を捉えていることが期待できる．そこで本稿では，BOW と複数の分散表現をアンサンブル学習により組み合わせる手法を提案する．

連絡先: 宅和晃志，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，takuwa@cmplx.cse.nagoya-u.ac.jp

2. アンサンブル学習

アンサンブル学習とは，複数の分類器を組み合わせて用いることで，分類精度を向上させる手法である．機械学習において，アンサンブル学習が有効な場合が多いことが報告されている [8]．

ここで，アンサンブル学習の一種であるバギングについて説明する．まず，用意された全学習サンプルから，ランダムに選んでは戻すという復元抽出を行い，複数種類の学習サンプルを作成する．それぞれの学習サンプルに対して分類器を適用し，全分類器の多数決をとることにより最終的な分類を行う．

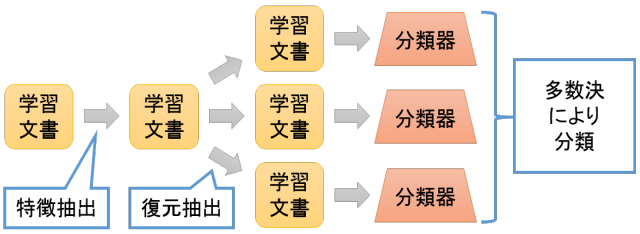
一方高橋ら [9] は，SVM においては，バギングなどのアンサンブル学習は有効性が高くないことを指摘し，各々の分類器が算出するクラス所属確率を用いたアンサンブル学習を提案している．ここで，クラス所属確率とは，分類対象である文書がそれぞれのクラスに所属する確率である．SVM におけるクラス所属確率は，Platt Scaling[10] と呼ばれる手法を用いることで算出される．

3. 提案手法

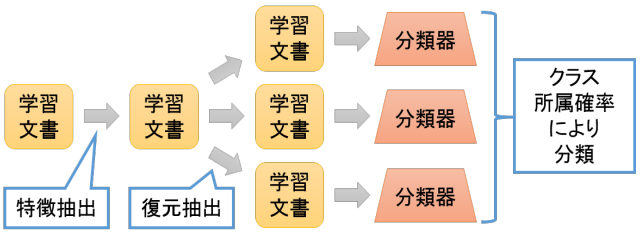
3.1 アンサンブル学習

提案手法では，2 章で述べたクラス所属確率を用いたアンサンブル学習を行う．様々な素性により特徴量を抽出した学習文書（学習サンプル）に対してそれぞれ分類器を適用する．具体的には，BOW，分散表現 A，分散表現 B，分散表現 C，… を素性として学習文書を作成し，それぞれの学習文書に対して分類器を適用する．次に，各分類器が算出するクラス所属確率に対して，各分類器に固有の重みをつけて足し合わせる．これにより得られた値の中で，最大となるクラスに分類を行う．重みの算出方法については，3.2 で述べる．

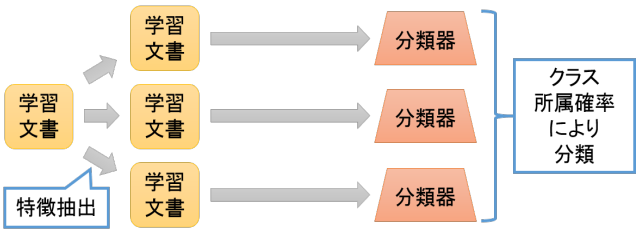
バギングや高橋らの手法では，単一の素性により特徴量を抽出しているのに対し，提案手法では複数の素性により特徴量を抽出している点で異なっている．(a) バギング，(b) 高橋らの手法，(c) 提案手法のイメージ図を図 1 に示す．



(a) バギング



(b) 高橋らの手法



(c) 提案手法

図 1: アンサンブル学習のイメージ図

3.2 重みの算出方法

初めに，分類器に重みをつける必要性について，例を用いて説明する．4 つの分類器 A,B,C,D を用いて，6 つの文書に対して分類を行い，各分類器の正誤状況が表 1 に示した通りであったとする．表 1 において，○は正しいクラスに分類できたことを示し，×は誤ったクラスに分類してしまったことを示す．

表 1 より，分類器 B,C,D はそれぞれ類似した判別傾向を持っており，分類器 A のみが異なった正誤判別を行っていることがわかる．この場合，各分類器が出力するクラス所属確率を，重みをつけずに足しあわせてしまうと，分類器 A の捉えている傾向と，分類器 B,C,D の捉えている傾向が 1:3 で足し合わされてしまうため，いわば分類器 B,C,D に重みがかかる形になる．正答率では，分類器 A〜D 全て 3/6 = 0.5 となっているため，これは最適な組み合わせ方とはいえない．この場合の最適な組み合わせ方は，分類器 A,B,C,D のクラス所属確率を 3:1:1:1 で足し合わせることであると考えられる．以上は極端な例ではあるが，各分類器の捉えている特徴が，それぞれ同等に活かされるような重みをつけて，足し合わせる必要があると考えられる．そこで，以下に示す方法で各分類器に対する重みを算出する．

表 1: 各分類器の正誤状況の例

	1	2	3	4	5	6
分類器 A	○	○	○	×	×	×
分類器 B	×	×	×	○	○	○
分類器 C	×	×	×	○	○	○
分類器 D	×	×	×	○	○	○

初めに，各分類器に対して点数という概念を考え，それぞれ初期の点数を 0 点とする．次に，学習文書に対する Leave-One-Out により，各学習文書に対する正誤状況を表 2 のように作成する（表 2 は学習文書が 6 つの例）．そして，各文書において正解した分類器に点数を与えていく．このとき，他の分類器の正誤状況によって与える点数を変化させる．

表 2: 学習文書における各分類器の正誤状況

	1	2	3	4	5	6
分類器 A	○	○	○	×	○	×
分類器 B	×	○	○	○	○	×
分類器 C	×	×	○	○	○	×
分類器 D	×	×	×	○	○	×

表 2 を用いて，与える点数について説明する．文書 1 では，分類器 A のみが正解をしているため，分類器 A に 1 点を加点する．文書 2 では，分類器 A,B の 2 つが正解をしているため，それぞれに 1/2 点を加点する．文書 3 では，分類器 A,B,C の 3 つが正解をしているため，それぞれに 1/3 点を加点する．文書 4 でも，3 つが正解をしているため，それぞれに 1/3 点を加点する．文書 5,6 では，全ての分類器が正解/不正解となっているため，点数は加点しない．最終的に，分類器 A,B,C,D のそれぞれの重みは，1.83, 1.17, 0.67, 0.33 となる．分類器 A は他の分類器では捉えることができない特徴（文書 1 で正解）を持っているため，重みが大きくなり，分類器 D は他の分類器との差別化ができていないため，重みが小さくなる．分類器 A と B のように，正答率は同じ 4/6 でも，捉えているものの違いで重みが異なる．

以上の方法で重みを算出することで，各分類器の捉えている傾向が同等に足し合わされるため，組み合わせる分類器の数を増やすことによる分類精度の向上が期待される．

4. 実験設定

4.1 分類対象データセット

実験で使用した文書分類データセットを表 3 に示す．これらのデータセットは，[11] の著者のウェブサイト [*1] からダウ

*1 http://web.ist.utl.pt/acardoso/datasets/

ンロードして利用できる．また前処理として，ストップワードの除去が既に行われている．

表 3: 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
20 Newsgroups	11293	7528	20
Reuters 21578	5485	2189	8

4.2 分散表現学習コーパス

実験では，表 4 に示す 5 種類のコーパスを用いて分散表現を学習した．表中の分類対象データセットは，20 Newsgroups を分類する場合には 20 Newsgroups の学習文書，Reuters 21578 を分類する場合には Reuters 21578 の学習文書をそれぞれ用いたことを意味する．これら分散表現学習用のコーパスは，分散表現 A を除き， word2vec 公式配布サイト [*2] で紹介されているものを用いた．分散表現学習時の word2vec 実行オプションは，分散表現 A では「-size 300 -min-count 0」，分散表現 C,D,E では「-size 300」を用いた．「-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り，「-min-count」は出現回数が何回以下の単語をストップワードとするかを指定するオプションである．分散表現 B は，学習済みの分散表現が word2vec 公式配布サイトで配布されているため，それを用いた．いずれの分散表現も，次元数は 300 である．

表 4: 分散表現学習コーパス

分散表現名	学習コーパス
分散表現 A	分類対象データセット
分散表現 B	Google News データセット
分散表現 C	英語 Wikipedia 全文
分散表現 D	UMBC ウェブコーパス
分散表現 E	One Billion Word Language Modeling Benchmark データセット

4.3 構築した分類器と素性

実験で用いた分類器の詳細を表 5 に示す．分散表現に対しては，正答率の期待できる SVM を用いたが，BOW に対しては，使用したコンピュータ性能の都合上，NaiveBayes 分類器を用いた．NaiveBayes 分類器はラプラススムージング（スムージング項 $k = 1.0$）を用いた．SVM は RBF カーネルを用い，ハイパーパラメーターである C と γ は，各分類器について学習文書におけるグリッド探索により求めた最適値を用いた．

表 5: 構築した分類器と素性

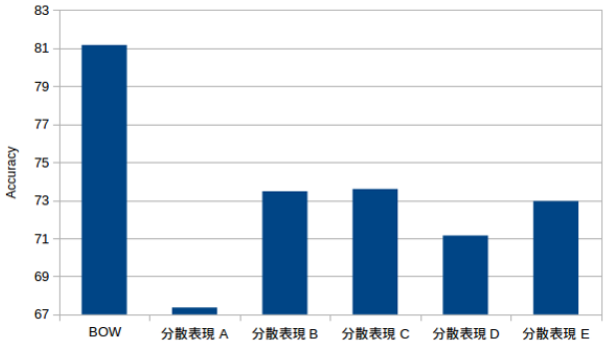
分類器	素性
NaiveBayes 分類器	BOW
SVM	分散表現 A
SVM	分散表現 B
SVM	分散表現 C
SVM	分散表現 D
SVM	分散表現 E

*2 https://code.google.com/archive/p/word2vec/

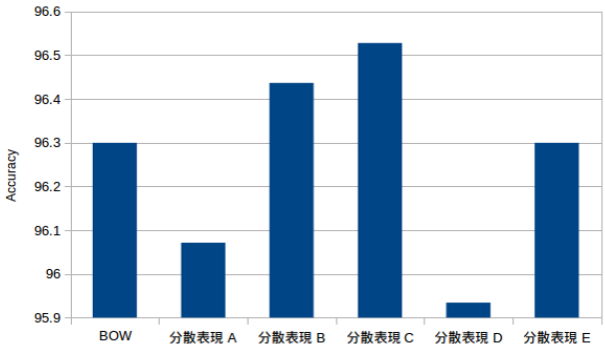
5. 実験

5.1 予備実験（分類器の性能確認）

各素性に対して構築した分類器の性能を確認することを目的とし，予備実験を行った．各分類器を単体で用いた場合の正答率を図 2 に示す．図から，各分類器単体では，20 Newsgroups では BOW が，Reuters 21578 では分散表現 B（Google News データセット），分散表現 C（英語 Wikipedia 全文）の正答率が高いことがわかる．



(a) 20 Newsgroups



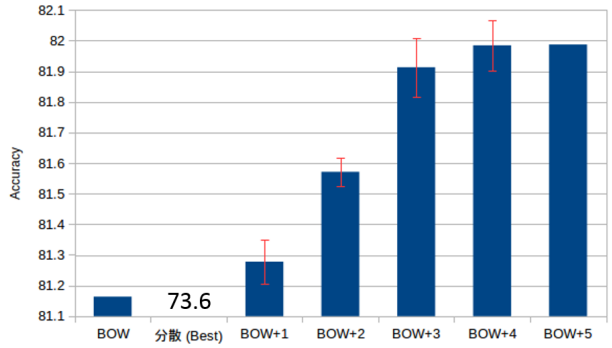
(b) Reuters 21578

図 2: 各素性単体の正答率

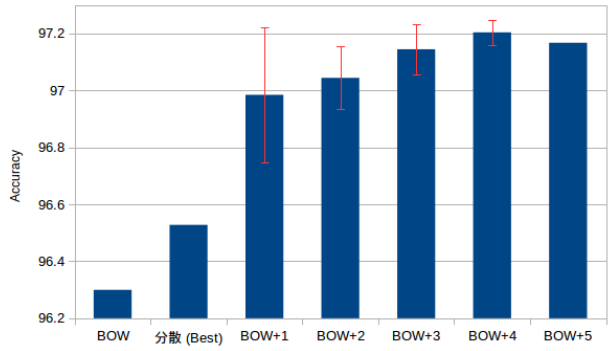
5.2 実験結果

従来手法（BOW，分散）と提案手法（BOW+1〜BOW+5）との比較実験を行った．実験結果を図 3 に示す．分散 (Best) は，図 2(a)(b) における分散表現 A-E の中で，最も正答率の高いものの正答率である．また例えば「BOW+2」とは，BOW を素性とする分類器と，分散表現を素性とする 5 種類の分類器のうち 2 つを加えた合計 3 つの分類器の組み合わせを行った手法を示す．この場合，分散表現を素性とする 5 種類の分類器から 2 つを選ぶため，${}_5C_2$ 通りの選び方がある．図 3 では，各選び方による正答率の平均値を示しており，標準偏差をエラーバーとして示している．

図 3 より，提案手法が従来手法に比べて正答率が高いことが確認できる．また，組み合わせる分類器の数を増やすことで正答率が向上することも確認できる．ただし，一部（Reuters 21578 の BOW+5）で，組み合わせる分散表現の数を増やし過ぎると，正答率が下がってしまうことがみられた．

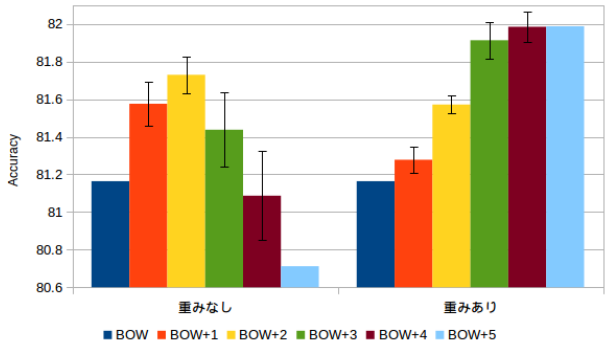


(a) 20 Newsgroups

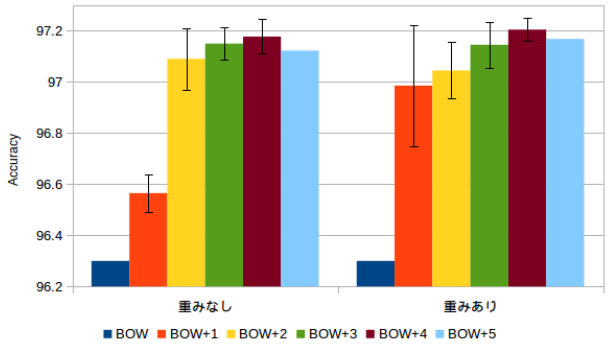


(b) Reuters 21578

図 3: 文書分類正答率



(a) 20 Newsgroups



(b) Reuters 21578

図 4: 重みの有無に対する正答率

3.2 で示した重みをつけた場合とつけない場合の比較実験の結果を，図 4 に示す．20 Newsgroups における重みなしの場合では，組み合わせる分散表現を増やすことにより大幅に正答率が下がっている（BOW+2〜BOW+5）ため，3.2 で示した方法により重みをつけることで，より適切に分類器の組み合わせが行われていると考えられる．

6. まとめ

本稿では，頻度情報を捉えることができる BOW と，単語間の意味的な関係を捉えることができる分散表現を組み合わせる方法について検討した．また，複数のコーパスからそれぞれ学習された複数の分散表現を利用し，BOW と複数の分散表現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した．さらに，学習データを用いて各分類器に重みをつける方法を提案した．文書分類実験を行い，提案手法の有効性を示した．

参考文献

[1] Domingos, Pedro, and Michael Pazzani. ”On the optimality of the simple Bayesian classifier under zero-one loss.” Machine learning 29.2-3 (1997): 103-130.

[2] Joachims, Thorsten. Text categorization with support vector machines: Learning with many relevant features. Springer Berlin Heidelberg, 1998.

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

[4] 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.

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

[6] Le, Quoc V., and Tomas Mikolov. ”Distributed representations of sentences and documents.” arXiv preprint arXiv:1405.4053 (2014).

[7] 澤井裕一郎, 吉川友也, 松本裕治. ”文書分類におけるサポートメジャーマシンの有効性.” 言語処理学会 第 21 回年次大会, 2015.

[8] Sebastiani, Fabrizio. ”Machine learning in automated text categorization.” ACM computing surveys (CSUR) 34.1 (2002): 1-47.

[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] Platt, John. ”Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods.” Advances in large margin classifiers 10.3 (1999): 61-74.

[11] Cachopo, Ana Margarida de Jesus Cardoso. Improving methods for single-label text categorization. Diss. Universidade T?cnica de Lisboa, 2007.