D-015

単語頻度を用いた文書分類と代表文書の抽出

Classification of Articles and Extraction of Representative Article in Each Article Class Using Frequencies of Words

木村 淳† 吉冨 康成‡ 田伏 正佳‡
Jun Kimura Yasunari Yoshitomi Masayoshi Tabuse

1. はじめに

近年，Web 上の情報は日々増え続けており，その情報量の多さのため，全てを個人が閲覧するのは困難である．しかし，それらは多くの情報量があるものの，各種メディア間で重複した内容を扱った文書が多々ある．そのような多くの情報源から効率的に情報を得るためには，情報の取捨選択が不可欠である．つまり，文書群の中から全体の内容を網羅しているような，いわば文書群の代表文書を抽出する方法が必要である．また，読む順番の参考とすることを目的としたランキングが，ユーザの効率的な情報把握に有効である．

文書分類を目的とした研究が盛んに行われているが(例えば，[1-13])，著者らの知る限り，各文書群の代表となる文書の抽出とその文書のランキング方法の報告はない．ユーザーが分類例示を行い，クラスタ代表(ベクトル)を更新しながら文書分類を行う研究 [5]が報告されているが，各クラスタの代表文書の抽出は行われていない．また，文書要約対象となる文書選択に際し，MMR[14]を使用する研究[12]が報告されているが，代表文書のランキングは行っていない．

そこで，本論文では，多量のニュース文書やブログ文書等の情報を閲覧する際に重複した内容の文書を読むことを避け，効率的に情報を得られるように，同一カテゴリの内容を扱う等の類似した内容の文書群の中から代表文書を自動抽出し，ユーザーにそれらの文書をランキング形式で提示する手法を提案する．

2. 提案手法

2.1 処理の流れ

初めに対象となる文書の表現の揺らぎの正規化を行った後に，文書の内容を表す特徴ベクトルを生成する．次に，各文書から生成した特徴ベクトルを基に Ward 法でクラスタリングを行い，文書分類を行う．そして，最後に分類した各クラスタから代表文書の抽出を行う．

2.2 文書の正規化と文への分割

一般的に，ニュース記事やブログ記事等の文書は数値やアルファベット等の文字列の扱いのフォーマットが均一ではないため，表現を統一するために文書の内容の正規化を行う必要がある．本法では文書に対して以下の 2 つの正規化処理を行う．

1. 全ての全角数字を半角の数字に置き換える．
例) 「２０１３」 => 「2013」

2. 全てのアルファベットを半角小文字のアルファベットに変換する．
例) 「Program」，「PROGRAM」を「program」に変換.

文字列としては異なる表現を用いていても，上記 2 つの正規化処理により，同一の表現として以降の処理を施すことができる．

そして，正規化を行った文書に対して，句点等により文に分割する．句点には一般的に文書に用いられると考えられる文字「。.!?！？」を利用した．これにより，1 つの文書を複数の文の集合と捉えることができる．

2.3 名詞の抽出

2.2 章記載の方法で分割して得られた各文に対して MeCab[15]で形態素解析を行い，名詞を抽出する．図 1 は，「インターネットを利用する．」という文を入力とした場合の MeCab での解析結果の出力例である．出力例からわかるように図 1 の例の場合は「インターネット」，「利用」という 2 つの単語が MeCab により名詞として抽出されており，これらの単語を以降の処理においても名詞として取り扱う．

```
インターネットを利用する。
インターネット　名詞,一般,*,*,*,*,インターネット,インターネット,インターネット
を　助詞,格助詞,一般,*,*,*,を,ヲ,ヲ
利用　名詞,サ変接続,*,*,*,*,利用,リヨウ,リヨー
する　動詞,自立,*,*,サ変・スル,基本形,する,スル,スル
。　記号,句点,*,*,*,*,。,。,。
EOS
```

図 1 「インターネットを利用する。」という文に対する MeCab での出力例

2.4 名詞の連結

2.3 章記載の方法で得られた各名詞に対して，接尾語かつ，直前に出現した名詞が数値である場合，連結して 1 つの名詞として取り扱う．

図 2 の例のように文中に「2013」，「年」と順に出現した場合，連結して 1 つの名詞「2013 年」として扱うといった処理を行う．これは，単位（例えば，「年」）と無

```
今年は2013年である。
今年　名詞,副詞可能,*,*,*,*,今年,コトシ,コトシ
は　助詞,係助詞,*,*,*,*,は,ハ,ワ
2013　名詞,数,*,*,*,*,*
年　名詞,接尾,助数詞,*,*,*,年,ネン,ネン
で　助動詞,*,*,*,特殊・ダ,連用形,だ,デ,デ
ある　助動詞,*,*,*,五段・ラ行アル,基本形,ある,アル,アル
。　記号,句点,*,*,*,*,。,。,。
EOS
```

図 2 「今年は 2013 年である。」という文に対す MeCab での出力例

† ジャストシステム，JustSystems Corp.
‡ 京都府立大学，Kyoto Prefectural University

関係に，数値（例えば，「2013」）を特徴ベクトルの要素として登録してしまうことを防ぐためである．

2.5 否定の属性の付与

文中に助動詞「ない」が出現した場合，同一の文中の名詞を否定の属性を付加した名詞として取り扱う．図 3 の例の場合，「ない」という助動詞と「企業」，「不正」という 2 つの名詞が抽出されている．この場合，「企業」と「不正」という 2 つの名詞にそれぞれ否定の属性を付与する．

```
あの企業は不正を行なってはいない。
あの	連体詞,*,*,*,*,*,あの,アノ,アノ
企業	名詞,一般,*,*,*,*,企業,キギョウ,キギョー
は	助詞,係助詞,*,*,*,*,は,ハ,ワ
不正	名詞,形容動詞語幹,*,*,*,*,不正,フセイ,フセイ
を	助詞,格助詞,一般,*,*,*,を,ヲ,ヲ
行なっ	動詞,自立,*,*,五段・ワ行促音便,連用タ接続,行なう,オコナッ,オコナッ
て	助詞,接続助詞,*,*,*,*,て,テ,テ
は	助詞,係助詞,*,*,*,*,は,ハ,ワ
い	動詞,自立,*,*,一段,未然形,いる,イ,イ
ない	助動詞,*,*,*,特殊・ナイ,基本形,ない,ナイ,ナイ
。	記号,句点,*,*,*,*,。,。,。
EOS
```

図 3　「あの企業は不正を行なってはいない。」という文に対する MeCab での出力例

2.6 特徴の抽出

2.3, 2.4, 2.5 章記載の方法で得られた名詞群中には，文字列として明らかに意味をなさないものも含まれていることも考えられる．名詞群中のそのような文字列を取り除くために，各名詞に対して以下の処理を順次行う．

＜処理＞
以下の 3 つの条件をすべて満たす名詞をリスト L に追加する．

＜条件＞
a. ひらがな 1 文字でない
b. カタカナ 1 文字でない
c. 記号のみで成り立つ文字列ではない

以上の処理により得られたリスト L を用いて，次章記載の方法で，その文書の特徴ベクトルを作成する．

2.7 特徴ベクトルの生成

2.6 章記載の方法で得られたリスト L を用いて文書の特徴ベクトルを生成する．まず，リスト L 中の名詞の種類数を次元数とする．そして，特徴量として「各名詞の文書中における出現回数/文書中の総名詞数」を用いる．これは，各文書の長さにはばらつきがあるため，特徴量がその影響を受けにくくさせるためである．以下に特徴ベクトルの要素の例を示す．

＜特徴ベクトルの要素の例＞
名詞, 特徴量
京都　0.018
大学　0.024
環境　0.007

2.8 文書分類

文書群中の各文書に対して，2.7 章記載の方法で特徴ベクトルを生成する．そうして得られた特徴ベクトル群に対してクラスタリングを行うことで，文書分類を行う．なお，クラスタリングには Ward 法を用いた．

2.9 代表文書の抽出

2.8 章記載の方法で得られたクラスタごとに，クラスタ中の文書群の重心ベクトルを求める．

そして，重心ベクトルと文書群の各特徴ベクトルとの，ユークリッド距離を算出し，その距離が最も近い特徴ベクトルをもつ文書をその文書クラスタ中の代表文書として抽出した．これにより，例えば，文書群を 4 つのクラスタに分類した際は，その文書群から 4 件の代表文書を抽出することになる．

本研究の代表文書の抽出手法によって得られた文書は，同一カテゴリに属する文書集合においては，特徴ベクトル空間での重心に一番近い文書であると期待できる．

2.10 代表文書のランキング方法

以下に示した処理を $i=1,2,\cdots,J$ の順に順次行い，それぞれ得られた代表文書群 M_i を $i=1$ から昇順にランキングしたものを代表文書ランキングとして，ユーザーに提示する．ここで，J は，順次行うクラスタリングにおけるクラスタ数の最大値である．なお，ランキング付きで抽出した文書群内の重複した文書については，最上位の文書のみを最終的なランキングに残す．

＜処理＞
2.8 章記載の方法で，文書群を i 個のクラスタに分類し，2.9 章記載の方法で，各クラスタの代表文書を抽出し，クラスタ内の要素数の降順でソートしたものを代表文書群 M_i とする．

3. 開発環境

開発環境，使用プログラミング言語を以下に示す．
・PC：DELL OPTIPLEX 780(CPU: Intel Core2 Duo CPU E8400 3.00GHz，RAM: 4.00GB)
・OS：Windows7 Professional
・使用プログラミング言語 Python 2.7.3

4. 性能評価実験

4.1 文書分類の性能評価

まず，2.8 章記載の文書分類の性能評価を行った．対象となる文書群には，Yahoo! Japan ニュース[16]上で「政治」を話題にしたニュース記事を 10 件(文書番号 1～10)，「競馬」を話題にしたニュース記事を 10 件(文書番号 11～20)の計 20 件の文書を 2013 年 1 月に収拾し，性能評価に供した．そして，2 つのクラスタとなるように文書分類を行ったところ，表 1 に示す結果が得られた．

表 1 記載の 2 つのクラスタ C_1，C_2 はそれぞれ「政治」の記事のみ，「競馬」の記事のみからなる．分類がそれぞれ

の記事カテゴリごとに行われていることから，文書分類が正確に行われたことがわかる．

表 1　各クラスに分類された文書の番号

クラスタ C_1	クラスタ C_2
1, 2, 3, 4, 5, 6, 7, 8, 9, 10	11, 12, 13, 14, 15, 16, 17, 18, 19, 20

4.2 代表文書抽出の性能評価

4.2.1 実験 1

文書群には Google ニュース[17]と Yahoo! Japan ニュース[16]において「大阪府 高校」をそれぞれの検索語句として指定して得られた 2013 年 1 月 22 日時点での検索結果上位 20 件の文書を用いた．文書群にはそれぞれ検索結果ランキングの順に 1-20 の文書番号とその文書のカテゴリを与えた．カテゴリは，文書群中に 2 種類以上の関連した内容の文書がある場合に，それらの文書の内容を表す名称を使用した．また，関連する文書が他にない文書は，「その他」というカテゴリに割り当てた．文書群に Google ニュースを用いた場合と，Yahoo! Japan ニュースを用いた場合の実験結果を以下に示す．当然ではあるが，カテゴリごとにクラスタリングされる保証はない．

(a)Google ニュース

文書群には，4 種類のカテゴリが存在し，表 2 のような構成となっていた．この文書群を用いて，2.10 章記載の方法で，$J=4$ としてランキングを行った結果を表 3 に示す．

表 2　文書群の構成 1

カテゴリ名	ラグビー	教育委員会	
文書番号	1, 9, 12, 18, 20	2, 4, 5, 11, 16, 17, 19	
カテゴリ名	スケート	遭難事故	その他
文書番号	3, 14	6, 7, 8, 15	10, 13

表 3　実験結果 1

代表文書の文書番号(ランキング順)
6, 1, 4, 3

代表文書 4 件が抽出され，順に「遭難事故」，「ラグビー」，「教育委員会」，「スケート」のカテゴリに属す文書となった(表 3)．元の文書群に存在した 4 種類のカテゴリの全てを代表文書で網羅した．

(b)Yahoo! Japan ニュース

文書群には，4 種類のカテゴリが存在し，表 4 のような構成となっていた．この文書群を用いて，2.10 章記載の方法で，$J=4$ としてランキングを行った結果を表 5 に示す．4 件の代表文書が抽出され，順に「教育委員会」，「センター試験」，「遭難事故」，「その他」のカテゴリに属す文書となった(表 5)．元の文書群に存在した 4 種類のカテゴリの内 3 種類を代表文書で網羅した．

表 4　文書群の構成 2

カテゴリ名	ラグビー	教育委員会	
文書番号	15, 19	2, 3, 4, 11, 14, 16, 17, 18, 20	
カテゴリ名	遭難事故	センター試験	その他
文書番号	8, 12, 13	9, 10	1, 5, 6, 7

表 5　実験結果

代表文書の文書番号(ランキング順)
18, 10, 8, 5

4.2.2 実験 2

文書群には Google ニュースと Yahoo! Japan ニュースにおいて「Microsoft」をそれぞれの検索語句として指定して得られた 2013 年 1 月 22 日時点での検索結果上位 20 件の文書を用いた．また，実験 2 で使用する文書群に関しても 4.2.1 章記載の実験 1 と同じ手法で文書番号とカテゴリを付与し，2.10 章記載の方法で，$J=4$ としてランキングを行った．

文書群に Google ニュースを用いた場合と，Yahoo! Japan ニュースを用いた場合の実験結果を以下に示す．

(a)Google ニュース

6 件の代表文書が抽出され，「その他」の記事が 2 件，「Windows8」，「Microsoft Security Essentials」，「Surface」，「その他」のカテゴリに属す文書となった(表 7)．元の文書群に存在した 3 種類のカテゴリの全てを代表文書で網羅した．

表 6　文書群の構成 3

カテゴリ名	Windows8	MS Essentials
文書番号	3, 5, 14	2, 6, 12
カテゴリ名	Surface	その他
文書番号	9, 11, 15	1, 4, 7, 8, 10, 13, 16, 17, 18, 19, 20

表 7　実験結果 3

代表文書の文書番号(ランキング順)
18, 4, 3, 6, 11, 20

(b)Yahoo! Japan ニュース

文書群は，1 種類のカテゴリのみが存在し，表 8 のような構成となっていた．この文書群を用いて，ランキングを行った結果を表 9 に示す．

表 8　文書群の構成 4

カテゴリ名	キャノン ITS	その他
文書番号	6, 9, 14	1, 2, 3, 4, 5, 7, 8, 10, 11, 12, 13, 15, 16, 17, 18, 19, 20

表 9　実験結果 4

代表文書の文書番号(ランキング順)
2, 9, 15, 13, 20, 11

6 件の代表文書が抽出され，順に「その他」，「キャノン ITS」，残り 4 件は「その他」のカテゴリに属す文書となった(表 9)．文書群中にカテゴリ「その他」に属す文書が多いため，本法の適用には向かない対象ではあったものの，カテゴリを形成している「キャノン ITS」からは 1 件代表文書が抽出された．

4.3 考察

4.3.1 実験結果について

4.2.1 章記載の実験 1 や 4.2.2 章記載の実験 2 の(a)のように，元の文書群のカテゴリがある程度定まっている場合に

関しては，本法の文書の分類が上手く機能しており，代表文書として各カテゴリの文書が重複することなく抽出できた．他方，4.2.2 章記載の実験 2 の(b)のように，文書群に「その他」に分類される文書が多数存在する場合は，代表文書だけでは文書群の内容を網羅することはできない．今後，適用例を増やして，本法の有効性をさらに確認する．

4.3.2 精度向上に向けて

本法の精度向上のために，シソーラスの利用が考えられる．特徴ベクトルの各要素について，シソーラスを利用することにより，類義語を同一の要素として扱うことができるようになるため，本法の精度向上につながると考えられる．また，これにより特徴ベクトルの次元圧縮が可能となり，処理速度の改善も期待できる．本研究では，クラスタリングには Ward 法を用いたが，精度向上の観点から，他のクラスタリング手法も検討する．

4.3.3 処理速度について

今回行った 2 つの実験ではそれぞれ 20 件ずつの文書を用いたが，この件数が大きくなるほど，代表記事を抽出する処理速度は遅くなっていくと考えられる．さらなる実用化に向けては，4.3.2 章で記載したシソーラスによる特徴ベクトルの次元圧縮による高速化以外にも，高速化方策を検討する必要がある．

4.3.4 代表文書の定義について

本論文では，カテゴリごとに分類した文書クラスタ内の重心に最も近い文書を代表文書として抽出しているが，代表文書の有用性あるいは適切性は，いかに人間の直感に沿った記事であるかどうかという点に尽きると考えられる．そのため，今後，本手法により抽出された文書が実際に人間が読む代表記事として相応しいのかどうかという正当性の評価方法の検討を行う必要がある．その際，代表文書の他の定義の仕方も合わせて検討する．

5. 結言

本研究では，文書群を複数のクラスタに分類し，それぞれのクラスタの中から代表文書を抽出する手法を提案した．

今後の展開を以下に示す．

- 類義語対応のためのシソーラス利用
- 特徴ベクトルの次元圧縮による処理の高速化
- 代表文書の正当性の評価方法の検討
- 代表文書の他の定義の仕方の検討

参考文献

[1] F. Can and E. A. Ozkarahan, "Computation of term/document discrimination values by use of the cover coefficient concept", Journal of the American Society for Information Science, Vol.38, No.3, pp.171-183, (1987).

[2]河合敦夫, “意味属性の学習結果にもとづく文書自動分類方式”, 情報処理学会論文誌, Vol.33, No.9, pp.1114-1122, (1992).

[3] 湯浅夏樹, 上田徹, 外川文雄, “大量文書データ中の単語間共起を利用した文書分類”, 情報処理学会論文誌, Vol.36, No.8, pp.1819-1827, (1995).

[4] 波多野賢治, 佐野綾一, 段一為, 田中克己, “自己組織化マップと検索エンジンを用いた Web 文書の分類ビュー機構”, 情報処理学会論文誌, Vol.40, No.SIG 3(TOD1), pp.47-59, (1999).

[5] 小林 優 , 吉高 淳夫 , 平川 正人, “特徴要素の重みを考慮に入れたクラスタ代表の洗練による文書クラスタリング”, 情処研報(デジタル・ドキュメント), Vol. 2002-DD-72, No. 28, pp.135-142, (2002).

[6] 高村大也, 松本裕治, “文書分類のための共クラスタリング”, 情報処理学会論文誌, Vol.44, No.2, pp.443-450, (2003).

[7] 高村大也, 松本裕治, “SVM を用いた文書分類と構造的帰納学習法”, 情報処理学会論文誌:データベース, Vol.44, No.SIG 3(TOD 17), pp.443-450, (2003).

[8] 深谷亮, 山村毅, 工藤博章, 松本哲也, 竹内義則, 大西昇, “単語の頻度統計を用いた文章の類似性の定量化”, 電子情報通信学会論文誌, Vol. J87-D-Ⅱ, No.2, pp.661-672, (2004).

[9] 堀田徹, “ブログの自動分類とカテゴリ内におけるブログ推薦方式の提案”, 情処研報(数理モデル化と問題解決), Vol.2008-MPS-72, No.39, pp.147-150, (2008).

[10] 別所克人, 内山俊郎, 内山匡, “学習データのクラスタリングを用いた文書分類”, 信学技報, Vol. OIS2008-85, pp.61-64, (2009).

[11] 馬場康夫, 新里圭司, 柴田知秀, 黒橋禎夫, “キーワード蒸留型クラスタリングによる大規模ウェブ情報の俯瞰”,情報処理学会論文誌, Vol.50, No.4, pp.1399-1409, (2009).

[12] 吉田稔, 中川裕志, 渋谷久恵, 前田俊二, “テキストマイニングによる機器異常診断支援の試み”, DEIM Forum 2012 F5-4, (2012).

[13] 鈴木浩子, 横本大輔, 牧田健作, 宇津呂武仁, 河田容英, 福原知宏, “文書集合の話題俯瞰手法に関する分析”, 言語処理学会第 18 回年次大会論文集, pp.543-546, (2012).

[14] H. Lin and J. Bilmes, “A class of submodular Functions”, for Document Summarization, ACL 2011, pp. 510–520, (2011).

[15] MeCab, http://mecab.sourceforge.net/

[16] Yahoo! JAPAN ニュース, http://headlines.yahoo.co.jp/hl

[17] Google ニュース, https://news.google.co.jp/