# 文書関連性を素性として追加した文書クラスタリング

佐々木 稔　新納 浩幸
茨城大学工学部情報工学科

## 1　はじめに

文書クラスタリングは，与えられた文書データの集合に対して，内容の類似している文書をまとめて，クラスタと呼ばれるいくつかの文書集合に自動的に分割するものである．これは文書集合にどのような内容が含まれているのか，いくつのトピックから成り立っているのかなどといった特徴を抽出することができ，新しい情報を発見するテキストマイニングにつながる重要な構成要素となっている．そのため，文書クラスタリングは Web データ，新聞記事，特許や論文など，様々な文書集合に対して応用されている．

文書クラスタリングを行う処理でまず最初に行うことは，文書の中からクラスタリングに必要な特徴を抽出する処理である．この抽出した特徴を利用して，文書間における特徴の分布が似ているペアをまとめるボトムアップクラスタリングや文書集合全体の二分割を繰り返し行うトップダウンクラスタリングが行われる．従来の文書クラスタリングにおいては，手がかりとなる特徴として，文書に含まれる単語が最もよく利用される．単語を抽出するためには，一般的に形態素解析などを利用して文書内から名詞や動詞が抽出される．このようにして抽出された特徴に対して統計的に類似性を比較するために，特徴の頻度や IDF などの重み付けを考慮した重要度などが計算され，特徴を素性とするベクトルを形成し，それらを索引語-文書行列として表現し，ベクトル空間モデルとして文書間での比較が行われる．

このように，従来の文書クラスタリングでは単語を素性として行っていたが，単語だけではなく，他の素性も利用したベクトル空間モデルに基づくクラスタリング手法や類似度計算手法も提案されている．例えば，Co-Clustering のように，はじめに素性として使う単語をまずクラスタリングして，いくつかの単語をまとめたクラスタを形成し，それを素性とするクラスタリング手法が存在する．この手法を用いることにより，文書クラスタリングの精度が向上したという報告がされている [2][4]．また，類似度の計算で使われるカーネルトリックは実際に単語をまとめることはしないが，単語間のなんらかの組合せを素性として大きな次元での内積計算が行われている [3]．この方法はサポートベクターマシンで一般的に利用され，最も性能の良い機械学習手法のうちのひとつとしてよく利用されている．さらに，Adaptive Sprinkling のように索引語-文書行列の主成分ベクトルを元の行列に追加することで，精度の高い特徴ベクトルが抽出できることが報告されている [1]．このように単語以外の素性を利用して，文書のクラスタリングや分類などの目的に対する性能を向上する研究が盛んに進められている．

このように， Co-Clustering やサポートベクターマシンは単語素性間の関連性を利用した素性の抽出を行っているが，これまでの研究においてクラスタリングをしたい文書集合と他の文書集合との関連性を素性として利用する手法は提案されていない．クラスタリング対象外の文書を素性として利用すると，関連のある文書間では関連度が高くなるので，ある対象外の文書に対して関連度の高いいくつかの文書はまとめることが可能である．そのため，クラスタリング精度を向上するためには文書間関連度も重要な手がかりとして利用可能であると考えられる．また，クラスタリングをする文書集合以外にもこれまでに文書多数の文書が作成され，少なからず関連のある文書が存在していると考えられる．例えば，新聞記事 1 か月分とその前の 1 か月分，同じ企業が以前出した特許など，関連する情報がいくつか存在する場合がある．そのため，文書間関連性を利用することで文書クラスタリング精度が向上すると考えられる．

本稿では，これまで単語の重要度を手がかりとしてクラスタリングを行っていたものを，他の文書の関連

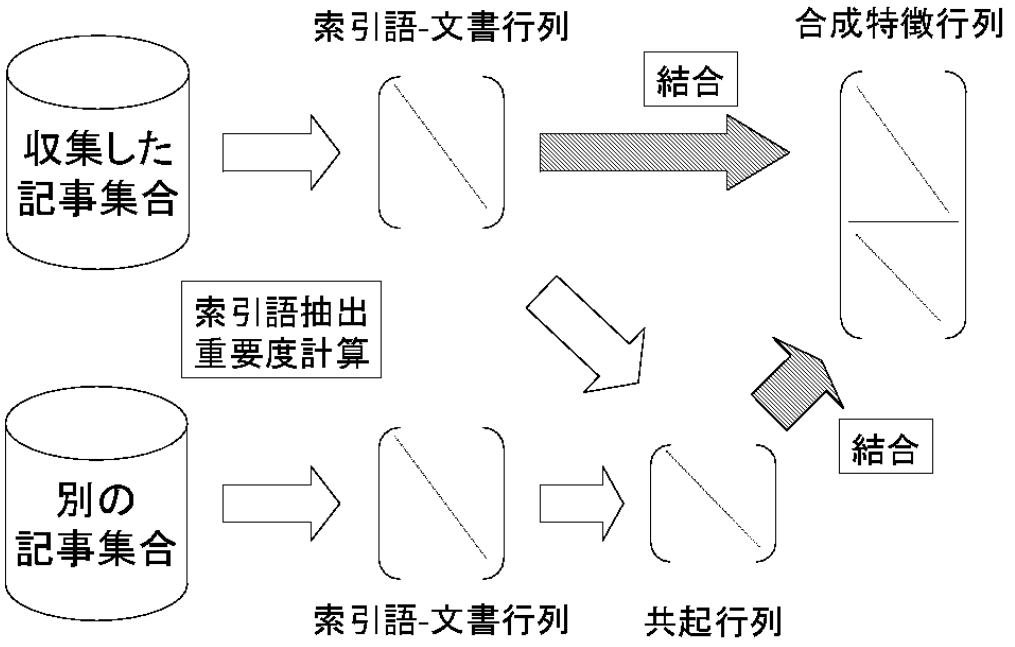


図 1: 文書関連度を追加した特徴行列作成の流れ

性を単語の重要度に加えて文書クラスタリングを行う手法を提案する．提案手法を用いた文書クラスタリング精度と用いない精度を比較して、どの程度の精度変化が出るかどうかを実験により比較した。

## 2　クラスタリングに使う素性

文書クラスタリングを行う際，文書内容を表現するために文書内に含まれる単語集合を抽出して，単語をクラスタリングの素性として利用する．このような単語集合を抽出することは，情報検索では索引付けと呼ばれ，索引語の重要度を計算することにより検索された文書に対してランキングが作成される [5]．クラスタリングについても索引付けと同様の処理で自動的に素性の抽出を行い，各素性の重要度計算を行う．

### 2.1　素性の抽出と重要度計算

クラスタリングを行う際に手がかりとなる素性として，文書から単語を抽出する．そのために，クラスタリングを行う文書集合に対してそれぞれ形態素解析を行い，文書を品詞ごとに単語分割を行う．本稿における実験では形態素解析器に MeCab[1] を利用した．その単語集合から数，代名詞，非自立，特殊，接尾辞を除く名詞と非自立を除く動詞の原形の抽出を行い，得られたすべての単語を素性として利用する．

抽出された単語集合に対して，各素性単語が文書内容についてどの程度の重要度を持っているか計算する．重要度計算には数多くの手法が提案されているが，本稿における実験では文書内での単語頻度 (TF) を利用した．TF の他にも重要度の計算手法には TFIDF や Log-Entropy などが存在する．本稿では，素性を追加した場合の性能を比較評価することが目的であるため，元の索引語-文書行列で使う重要度計算はどのようなものを利用しても構わないこととした．そのため，単語素性の重み付け手法は重要度計算手法の中で最も簡単に計算が可能な TF を利用することとする．しかし，TF を利用した場合，追加する素性としてコサイン類似度を用いていることもあり，追加する素性の値と範囲が大幅に異なる場合がある．そのため，元の索引語-文書行列を作成する際，行列に対して正規化を行い，文書ベクトルの長さを 1 に統一する処理を行う．このように，素性を追加する効果が現れるように，追加する素

[1] http://mecab.sourceforge.net/

性値の範囲は元の行列と似た値の範囲としている．

### 2.2 特徴として追加する素性

本稿で提案するクラスタリング手法では，索引語-文書行列に追加する素性として文書間の類似度行列を追加することとする．この類似度行列を追加するためには他の文書集合を用意する必要がある．用意した他の文書行列に対して元の行列と同様にして索引語-文書行列を計算する．元行列の各列と追加文書行列の各列に対して類似度計算を行い，追加文書を行ベクトルに，元の文書を列ベクトルに対応するような類似度行列を求める．このとき，類似度計算にはコサイン尺度を利用した．ただし，ベクトル間で内積計算を行う必要があるので，行列の要素がどの単語に対応するのかを一致するように対応付けておく．このようにして求めた類似度行列を元の索引語-文書行列の下に追加し，単語と文書間類似度を要素として持つ合成文書ベクトルを作成する．これまでの一連の処理の流れを図 1 に示す．

## 3 クラスタリング

クラスタリングの対象となる文書数 $n$ とし，そこから抽出される単語数を $m$，関連度を追加するために用いる他の文書集合に含まれる文書数を $l$ とすると，関連度を追加した行列は $(m+l) \times n$ の合成特徴行列となる．この行列に対してクラスタリングを行い，内容の似ている文書をまとめる．

索引語-文書行列に対してクラスタリングを行う手法はこれまでに数多くの研究が進められている．このクラスタリング手法には分割型と凝集型があり，分割型は文書集合からいくつかの内容のまとまりであるクラスタを求め，これをクラスタの変化がなくなるまで繰り返す処理を行う．また，凝集型はひとつの文書をひとつのクラスタに対応させ，似ているふたつの文書をひとつにまとめることを，与えられたクラスタ数になるまで繰り返す処理を行う．

本稿では，以上のように数多く提案されているクラスタリング手法の中で，$k$ 平均クラスタリングと似た Co-Clustring 手法である Minimum Sum-Squared Residue Co-clustering を利用して評価実験を行った [2]．今回の実験では文書関連性を示す素性を追加した場合のクラスタリング性能を評価することが目的であるため，クラスタリング手法はひとつに限定して評価することとした．しかし，提案手法が利用するクラスタリング手法と関係があり，高精度な文書クラスタリングが可能となることも考えられるため，いくつかのクラスタリング手法を利用した実験については今後の課題とする．

このクラスタリング手法を利用して，合成特徴行列のクラスタリングを行う．このクラスタリング手法は列ベクトルと行ベクトルのそれぞれに対して交互にクラスタリングを行うが，今回の実験では列ベクトルである文書のクラスタリングを 1 回のみ行い，得られたクラスタリング結果を評価することとした．

表 1: 記事集合の記事数とクラスタ数

| 記事集合 | 記事数 | クラスタ数 |
|---|---|---|
| A | 121 | 13 |
| B | 119 | 12 |
| C | 121 | 13 |

## 4 提案手法の評価実験

本節では，提案した手法の有効性を検証するため，新聞記事を利用してクラスタリング実験を行った．この実験を行うにあたり，3 つの記事集合を対象として，そのうちの 2 つの集合を組み合わせることで 6 種類の評価データを作成し，評価を行った．実験の評価方法はクラスタリングを行う各記事集合に対して，ベースラインとして元の索引語-文書行列，提案手法として合成特徴行列におけるクラスタリングをそれぞれ行い，クラスタ内における内容の異なる記事数の合計を求め，合計エラー数を比較した．

### 4.1 データ

評価データには，Google ニュース[2] において同じ内容の記事としてまとめられた記事集合を利用した．3 つの時点において，Google ニュースのリンクを利用して，様々なジャンルにおける 12 種類程度の類似した内容の文書を抽出して，それぞれ 120 件程度の文書集合を

[2] http://news.google.co.jp/

構成した．また，同じ記事内容について書かれているものに対しては同じクラスタに属するものとして，手作業により分類を行い，それを正解データとした．表 1 に，評価データとなる 3 種類の記事集合について，記事数とクラスタ数を示す．表 1 における記事集合は，時系列で A, B, C の順に新しくなっている．

### 4.2 実験結果・考察

表 2 に実験を行った結果を示す．表 2 の各列はそれぞれクラスタリングを行う文書集合，何も追加しないベースラインとしてのエラー数と記事集合 A, B, C をそれぞれ追加した場合のエラー数を表している．ただし，今回の実験では異なる文書集合を追加することを目的としているため，同じ文書集合の関連度を追加する実験は行っていない．

実験の結果，異なる文書集合の関連度を素性として追加することにより，どのような組み合わせにおいてもエラー数が減少した．この結果より，文書間関連度を素性として追加することで，エラー数の減少幅は異なるもののクラスタリング精度が向上することを確認することができた．また，追加する記事集合について実験結果を分析すると，記事集合 B に A を追加する，または，記事集合 C に B を追加する実験において，エラー数の減少が顕著に現れた．これはクラスタリングを行う記事集合に対して関連のある記事が多く含まれていることが原因として挙げられる．そのため，記事集合 C では，より古いデータになるにつれてエラー数が減少しなかったと考えられる．しかし，新しいデータを追加した場合は，続報記事でも新しい内容が含まれ，さらに最新記事も多いことから，それほどエラー数の減少が起こらなかったのではないかと考えられる．そのため，時系列データのテキストマイニングや類似特許のクラスタリングなどといった，関連する文書を含むデータ集合に対して効果的なクラスタリング手法となると考えられる．

## 5 おわりに

本稿では，これまで単語の重要度を手がかりとしてクラスタリングを行っていたものを，他の文書の関連性を単語の重要度に加えて文書クラスタリングを行う手法を提案した．実験によりクラスタリング精度を比較した結果，異なる文書集合の関連度を素性として追加することにより，クラスタリング精度が向上することを確認することができた。また，関連の強い文書集合を追加することにより，クラスタリング精度が効果的に向上することが可能であることも分かった．

表 2: 各記事集合におけるエラー数の比較

| 記事集合 | 追加なし | A 追加 | B 追加 | C 追加 |
|---|---|---|---|---|
| A | 39 | - | 37 | 24 |
| B | 30 | 12 | - | 29 |
| C | 31 | 28 | 22 | - |

今後の課題としては，クラスタリングを行う際の索引語の重み付け手法やクラスタリング手法を変化させて，提案手法を利用することによる効果の違いを分析し，愛称の良いクラスタリング手法が存在するかどうかを分析することが挙げられる．また，索引語-文書行列に対して関連度を追加する際，効果的なクラスタリングを行うことができるための数値的なバランスを考慮する必要がある．追加する関連度の範囲が小さい場合は，追加する効果があまりなく，大きすぎると影響が強く出ることが考えられるため，有効な追加方法についてより詳細な分析を行いたい．

## 参考文献


[1] Sutanu Chakraborti, Rahman Mukras, Robert Lothian, Nirmalie Wiratunga, Stuart Watt, and David Harper. Supervised latent semantic indexing using adaptive sprinkling. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI07)*, pages 1582–1587, 2007.

[2] Hyuk Cho, Inderjit Dhillon, Yuqiang Guan, and Suvrit Sra. Minimum sum squared residue co-clustering of gene expression data. In *Proceedings of the Fourth SIAM International Conference on Data Mining*, 2004.

[3] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[4] Inderjit S. Dhillon, Subramanyam Mallela, and Dharmendra S. Modha. Information-theoretic co-clustering. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 89–98, New York, NY, USA, 2003. ACM.

[5] 北 研二, 津田 和彦, 獅々堀 正幹. **情報検索アルゴリズム**. 共立出版, 2002.