忘却の概念に基づくクラスタリング手法の改良方式

An Improved Approach to the Clustering Method Based on Forgetting Factors

石川 佳治♦ 北川 博之♦

Yoshiharu ISHIKAWA Hiroyuki KITAGAWA

ネットワーク上で配信されるニュース記事や，ディジタル図書館において時系列的に蓄積される文書などのさまざまなオンライン情報を要約したり，それらの中から適切な文書を選択したりするために，クラスタリングは有用な手法である．オンライン環境では，ユーザは一般に新規性の高い文書に対して興味を有することを考慮し，著者らは忘却（forgetting）の概念を導入した文書クラスタリング手法を提案している．忘却の概念に基づく文書類似度をクラスタリングに用いることで，新しい文書ほどよりクラスタリングの結果に影響を持つことになる．本稿では，これまで本研究で提案したクラスタリングのアルゴリズムを，*K*-means クラスタリング法をもとに改良するアプローチについて述べる．

Clustering plays important roles in various on-line applications such as extraction of useful information from news feeding services and selection of relevant documents from incoming scientific articles in digital libraries. In on-line environments, users generally have interests on newer documents than older ones and have no interests on obsolete old documents.

Based on this observation, we have proposed an on-line document clustering method that incorporates the notion of a *forgetting factor* to calculate document similarities. The idea is that every document gradually losses its weight (or memory) as time passes according to this factor. Since our method generates clusters using a document similarity measure based on the forgetting factor, newer documents have much effect on the resulting cluster structure than older ones. In this paper, we extend our clustering method by using the *K*-means clustering algorithm as its basis. The new algorithm has clear semantics and supports incremental updates of cluster structures.

1. はじめに

Webやインターネット上のニュースサービスなどの普及により，今日ではネットワークを介して大量の文書データがユーザに配信されている．時々刻々と配信される莫大な情報から必要な情報を抽出する労力は多大であるため，配信された文書集合の中から有用な文書を選択する情報フィルタリングや，文書の要点を抜き出す文書要約手法が重要な研究分野となっている．近年ではそれらに加えて，ニュース記事などからのトピックの抽出と追跡（topic detection and tracking, TDT）も着目を浴びている[1,2]．このような応用において，**文書クラスタリング**（document clustering）は情報の要約・抽出のための基盤技術として利用される．

インターネット上のニュース記事に配信時刻を対応付けできるように，ネットワーク上で配信される文書データには，それに時刻を対応付けできるものが多く存在する．そのような文書のことをここでは**時系列文書**と呼ぶ．時系列文書は，その対応する時刻が新しいほど一般に最近のトピックに関する情報を含んでいると考えられる．よって，文書のクラスタリングを行う場合に，文書の内容だけでなく文書が対応する時刻も考慮してクラスタリングを行えば，より精度のよい文書クラスタリングが実現可能であると考えられる．

このようなアイデアに基づき，我々は忘却の概念に基づくクラスタリング手法を提案した[3]．その基本的なアイデアは，文書の時間的な忘却の概念を導入し，文書が古くなるほどその価値が減少するというモデル化を行い，文書類似度を導出する点にある．このモデル化に基づいて導出した文書間の類似度を用いると，文書は古くなればなるほど他の文書との類似度が減少することになる．これは，古い文書を「忘却」していると考えることができ，このような類似度をクラスタリングに用いることで，新規性の高い文書を中心にクラスタリングを行うことが可能となる．

文書の類似度とは別に，どのようなクラスタリングアルゴリズムを用いるかという選択肢は，クラスタリングの効率やクラスタリング結果の質を考える上で重要な要素である．論文[3]では，我々はCanらにより提案されたインクリメンタルな文書アルゴリズム [4]を拡張し，忘却の概念に基づく類似度を導入し，新たなクラスタリング手法の開発を行った．[4]の手法は，次々と文書がストリーム的に追加される，本研究が想定する状況に適したものであるが，用いられるクラスタリングの良さを表す指標に不明確さが存在し，手法としての妥当性に問題があった．そこで我々は，論文[5]において，大量の文書データのクラスタリングのために提案されたScatter/Gather法[6]に基づくアルゴリズムを提案した．一部のデータをサンプリングし高コストであるが精度のよい階層的クラスタリングをまず適用し，作成された初期クラスタに残りの文書をマージするのが基本的なアプローチである．この手法については，クラスタリングの基準が明確であるという利点があったが，クラスタリングに要する時間が大きく，また，文書の追加に応じてインクリメンタルにクラスタを更新することができず，毎回クラスタリング処理を実行しなければならないという欠点があった．

このような問題点を踏まえ，本稿では，クラスタリング手法として一般的な手法の一つである***K*-means法**に対し，忘却の概念に基づく類似度を導入した手法を提案する．*K*-means法ではクラスタリングの目標を目的関数の最小化と捉えることができるため，[3]で問題となったようなクラスタリングの指標の妥当性の問題が解決できる．また，文書の追加に応じてクラスタリング結果をインクリメンタルに更新可能であるため，[5]で発生した更新コストの問題も解決できると考えられる．本稿ではそのアイデアを中心に，提案手法の概要について述べる．

2. 忘却の概念に基づく文書類似度

2.1 影響力の逓減モデル

まず，本研究で用いる文書類似度を導出する上で基礎となる影響力の逓減モデルについて簡単に説明する．現在の時刻

♦正会員 筑波大学電子・情報工学系
{ishikawa, kitagawa}@is.tsukuba.ac.jp

を $t=\tau$ とする．ネットワークを介して配信され，文書リポジトリに現在格納されている文書を $d_i\,(i=1,\ldots,n)$ とし，それぞれの入手時刻に対応するタイムスタンプ（例：新聞記事ならば発行日など）を $T_i\,(T_i \le \tau)$ とする．ここで各文書に対し，その文書のタイムスタンプと現在の時刻との間の関係で定まる**影響力**（influence value）の値を以下のように定義する．

$$dw_i|_\tau = \lambda^{\tau - T_i}\ (0<\lambda<1) \tag{1}$$

文書の影響力のことを文書の**重み**（weight）と呼ぶこともある．この式により，文書 d_i はその入手時刻 $t=T_i$ において最大の重み $dw_i=1$ をとり，時間が経つにつれ重みが次第に減少し，最後には 0 に限りなく近づくことになる．λ は影響力の逓減の度合いを表すための定数であり，**忘却係数**（forgetting factor）と呼ぶ．この値が小さいほど文書の重みの逓減の度合いが大きくなることになる．

2.2 文書類似度の導出

次に類似度の導出に移る．まず，文書リポジトリからの文書の選択確率を以下の式で主観確率（subjective probability）として定義する．

$$\Pr(d_i) = dw_i / tdw \tag{2}$$

ただし

$$tdw = \sum_{l=1}^{n} dw_l \tag{3}$$

である．つまり，文書はそれが入手された時点では $1/tdw$ という高い確率で選択されるが，時間が経つにつれてその選択確率が 0 に近づくことになり，古い文書が忘却されるという現象を表現することになる．

ここで文書 d_i が与えられたとき d_j が想起される確率 $\Pr(d_j|d_i)$ を，以下のように近似する（m は索引語数）．

$$\begin{aligned}\Pr(d_j \mid d_i) &= \sum_{k=1}^{m} \Pr(d_j \mid d_i, t_k)\Pr(t_k \mid d_i) \\ &\approx \sum_{k=1}^{m} \Pr(d_j \mid t_k)\Pr(t_k \mid d_i)\end{aligned} \tag{4}$$

この近似は，確率的情報検索モデルで式の簡略化のために用いられる仮定に基づくものである．これより，文書 d_i, d_j の共起確率は以下のように与えられる．

$$\begin{aligned}\Pr(d_i, d_j) &= \Pr(d_j \mid d_i)\Pr(d_i) \\ &\approx \Pr(d_i)\sum_{k=1}^{m} \Pr(d_j \mid t_k)\Pr(t_k \mid d_i)\end{aligned} \tag{5}$$

本手法ではこの確率を文書 d_i, d_j 間の類似度として扱う．

$$sim(d_i, d_j) = \Pr(d_i, d_j) \tag{6}$$

すなわち，文書リポジトリから同時に2つの文書を取り出す際に，d_i, d_j のペアが抽出される確率を両者の類似度とする．

上で示した式で用いられる $\Pr(t_k|d_i)$ については，単純に出現頻度の比率を用いて

$$\Pr(t_k \mid d_i) = f_{ik} / \sum_{l=1}^{m} f_{il} \tag{7}$$

と定義する．f_{ik} は文書 d_i 中に索引語 t_k が出現する回数を表す．$\Pr(d_j|t_k)$ については，ベイズの定理を用いて

$$\Pr(d_j \mid t_k) = \Pr(t_k \mid d_j)\Pr(d_j) / \Pr(t_k) \tag{8}$$

となる．ただし，n を文書の総数としたとき，

$$\Pr(t_k) = \sum_{i=1}^{n} \Pr(t_k \mid d_i)\Pr(d_i) \tag{9}$$

である．以上の式を組み合わせることで，文書どうしの類似度を計算することが可能となる．

なお，ここで

$$\begin{aligned} tf_{ik} &= f_{ik} \\ idf_k &= 1/\sqrt{\Pr(t_k)} \\ len_i &= \sum_{i=1}^{m} f_{il} \end{aligned} \tag{10}$$

および

$$\vec{d}_i = (tf_{i1}\cdot idf_1, tf_{i2}\cdot idf_2, \ldots, tf_{im}\cdot idf_m) \tag{11}$$

とおくと，類似度式は

$$sim(d_i, d_j) = \Pr(d_i)\Pr(d_j)\frac{\vec{d}_i \cdot \vec{d}_j}{len_i len_j} \tag{12}$$

となる．すなわち，提案する類似度は *tf-idf* 法の拡張であることがわかる．

3. *K*-means 法に基づくクラスタリング

3.1 *K*-means 法

K-means 法は広く用いられているクラスタリング手法の一つであり，繰り返し処理により，初期状態のクラスタに洗練を行い，質の高いクラスタリング結果を生成することを目的としている．一般的なアルゴリズムは**図 1** のようになる．

1. K 個の文書をランダムに選択して，K 個の初期クラスタを生成する．
2. （残りの）各文書を各クラスタ代表と比較して，最も適切なクラスタに割り当てる．
3. クラスタへの割り当て結果に変化がない（または十分にクラスタ割り当てが収束した）ならば終了．そうでなければ，各クラスタの代表を再計算し，ステップ 2 に戻る．

図 1 一般的な *K*-means 法のアルゴリズム
Fig. 1 General Algorithm of *K*-means Method

このアルゴリズム自体は単純なものであるが，(a) クラスタ代表をどのように定義するか，(b) ステップ 2 において最も適切なクラスタをどのような基準で選択するか，(c) ステップ 3 において，クラスタリングの収束条件としてどのような基準を用いるか，などでさまざまなバリエーションが考えられる．

3.2 クラスタリングの指標の定義

クラスタリングの結果の良し悪しを測るための基準として，まず，クラスタ $C_p\,(1 \le p \le K)$ 中の文書の平均類似度を以下のように定義する．

$$avg\_sim(C_p) = \frac{1}{|C_p|(|C_p|-1)} \sum_{d_i \in C_p} \sum_{d_j \in C_p, d_j \neq d_i} sim(d_i, d_j)$$

(13)

ここで $|C_p|$ はクラスタ C_p の要素数である．この式では，クラスタ中のすべての文書のペアについて類似度を求め，それらの総和をとり，それを組合せの総数に比例する値で割って平均化している．これにより，クラスタ中の文書が互いに似ていれば似ているほど，平均類似度の値が大きくなることになる．次に，この平均類似度を用いて，クラスタリング結果の良さを与える指標を以下のように定義する．

$$G = \sum_{p=1}^{K} |C_p| \cdot avg\_sim(C_p) \qquad (14)$$

すなわち，各クラスタについて平均類似度とクラスタの要素数の積を計算し，それらの総和をとったものがクラスタリングの指標となる．直感的には，包含する文書が互いに類似しているような文書を多数含むようなクラスタ分割が得られた際に，G の値が大きくなる．3.1 節のアルゴリズムの説明では，各クラスタについてクラスタ代表を計算しておき，それを用いてクラスタへの割り当てを進めるようになっているが，この指標を用いれば，定義上はクラスタ代表を用いる必要はない．ただし後述のように，実際には，計算の効率化のテクニックとしてクラスタ代表を利用する．

なお，上式においてクラスタ数 $|C_p|$ を掛けずに平均類似度の和だけを用いる場合には，互いに強く類似しているが非常に小規模なクラスタが K–1 個，互いの類似度が小さいが大規模なクラスタが 1 個生じ，良好な結果とならない場合が多いことが，予備実験によって明らかとなっている．クラスタ数を掛けることで，要素数が非常に小さいクラスタが生成されることを排除できる．

3.1 節で示した K-means 法の処理のステップ 3 において，ここで定義した G を用いる．繰り返しが行われるたび，一般的には G の値は増加するが収束に向かうので，G が収束した時点での結果をクラスタリングの結果として採用する．K-means 法の性質により，この結果は G を最大とする解ではなく極大にする解であることに注意が必要であるが，明快な基準が得られたことになる．

3.3 提案アルゴリズム

次に，3.1 節で述べた K-means 法アルゴリズムのステップ 2 について考える．本手法では，各文書を適切なクラスタに割り当てるため，前節で定義した指標 G を用いる．すなわち，ある文書を割り当てるクラスタを決定する際に，G の増加に最も貢献するようなクラスタを選択する．

なお，文書によっては，どのクラスタに追加しても G を減らしてしまうものが存在する．そのような文書はいわゆる外れ値（outlier）であり，どのクラスタに入れてもそのクラスタの平均類似度を落としてしまうという性質がある．本研究では忘却の概念を導入した類似度を用いているが，この類似度では古い文書は他のどの文書に対しても類似度が小さくなるという外れ値の傾向を示すため，このような文書が多く発生する傾向にあり，この問題への対策は重要である．ただし，古い文書を忘却したいという本研究のアイデアを考慮すれば，このような外れ値は積極的に外れ値として扱う方が妥当であると考えられる．このような点をふまえ，指標 G の増加につながらない文書については，どのクラスタにも追加せず，外れ値リストで管理することにする．

以上の考察に基づき，本研究で提案する K-means 法は**図 2** のようになる．

アルゴリズムは初期化処理と繰り返し処理からなる．繰り返し処理では，先に述べたように各文書を入れるべき適切なクラスタを指標 G に基づいて決定する．適切なクラスタがない場合は，文書を外れ値リストに入れる．なお，いったん外れ値リストに入れても，次の繰り返し処理のステップ 1 では再び文書を検討対象とする．クラスタ内容の変化に伴い，次回には外れ値にならない場合が生じるためである．

初期化処理

1. K 個の文書をランダムに選択して，K 個の初期クラスタを生成する．
2. 各クラスタのクラスタ代表を計算する．
3. 指標 G を計算する．

繰り返し処理

1. 各文書 d について以下の処理を行う．
   a) その文書を各クラスタに追加した際の G の値を計算する．
   b) G の値を最も増加させるクラスタに d を追加する．どのクラスタに追加しても G が増加しない場合，d を外れ値リストに追加する．
2. 各クラスタのクラスタ代表を再計算する．
3. 指標 G の値を再計算し，G_{new} とおく．
4. 前回の G の値を G_{old} としたとき，$(G_{new} - G_{old}) / G_{old} < \delta$ が成立したらアルゴリズムを終了する．δ はあらかじめ与えられた定数である．
5. 繰り返し処理のステップ 1 に戻る．

図 2 提案する K-means 法のアルゴリズム
Fig. 2 Proposed Algorithm of K-means Method

このアルゴリズムに従えば，一般には G の値は繰り返しのたびごとに大きい値に更新され，最終的には収束する．ただし例外的なケースとして，G が減少する場合も発生する．繰り返し処理のステップ 1 で各文書の割り当てを計算するときに前回計算されたクラスタ代表を用いるが，ステップ 2 でクラスタ代表を再計算すると，前回のクラスタ代表と値が若干変化するためである．このような現象は，クラスタリングが収束に達する近辺で，振動のような形で発生することがある．そのため，ステップ 4 の収束条件の判定では，増加量が負になった場合は前回のクラスタリング結果を解とするなどの工夫を行うことになる．

3.4 クラスタ代表を用いた効率的計算法

先に述べたように，(14)式の G をクラスタリングの指標として K-means 法を適用する場合，クラスタ代表は本質的には不要である．なぜなら，クラスタへの文書の割り当てが決まれば，(14)式の値は計算可能であるためである．しかし，この計算には各クラスタ中のすべての文書について総当りで類似度を計算しなければならないため，大量の計算が発生する．図 2 に示したアルゴリズムの繰り返し処理のステップ 1 において，各文書においてクラスタごとに G の計算が発生するため，1 回の繰り返しにおいて nK 回 G を求めることになる（n は文書数である）．コストの高い G の計算を頻繁に行うため，オーバヘッドが大きい．

そこで，クラスタ代表を用いた効率的な G の計算方式を以下に示す．これは[6]の Scatter/Gather の論文などで示されたアイデアを拡張したものである．m を索引語の総数とする．クラスタ C_p のクラスタ代表のベクトルを

$$\vec{c}_p = [c_1^p, c_2^p, \ldots, c_m^p] \qquad (12)$$

で定義する．ただし，$1 \le k \le m$ について

$$c_k^p = \sum_{d_i \in C_p} \frac{\Pr(d_i) \cdot tf_{ik} \cdot idf_k}{len_i} \quad (13)$$

である．ここで，クラスタ C_p, C_q のクラスタ代表間の類似度を

$$cr\_sim(C_p, C_q) = \sum_{k=1}^{m} c_k^p c_k^q \quad (14)$$

と定義する．ここで，クラスタ C_p どうしのクラスタ代表類似度 $cr\_sim(C_p, C_p)$ の式を展開すると，

$$cr\_sim(C_p, C_p) = |C_p|(|C_p|-1) \cdot avg\_sim(C_p) + ss(C_p) \quad (15)$$

となる．ただし，

$$ss(C_p) = \sum_{d_i \in C_p} sim(d_i, d_i) \quad (16)$$

である．これにより，平均類似度の式は

$$avg\_sim(C_p) = \frac{cr\_sim(C_p, C_p) - ss(C_p)}{|C_p|(|C_p|-1)} \quad (17)$$

と変形できる．

ここで，2つの共通要素を持たないクラスタ C_p, C_q の和集合をとったクラスタを $C_r = C_p \cup C_q$ とすると，特に C_q が単一文書からなるクラスタ $C_q = \{d_q\}$ の場合，

$$avg\_sim(C_r, C_r) = (cr\_sim(C_p, C_p) + 2cr\_sim(C_p, C_q) - ss(C_p)) / (|C_p|(|C_p|+1)) \quad (18)$$

となる（導出の詳細は省略）．すなわち，既存のクラスタ C_p にある文書 d_q (= C_q)を追加したときの avg_sim の計算には，2つのクラスタ代表の類似度計算 $cr\_sim(C_p, C_q)$ のみをその時点で行えばよいことになる．なぜなら，$cr\_sim(C_p, C_p)$, $ss(C_p)$, $|C_p|$ は，クラスタ C_p が作られた時点であらかじめ計算し保持しておけばよいためである．これにより，あるクラスタにある文書を追加したときに avg_sim の値がどうなるかを，低コストの処理で計算できることになる．

同様に，既存のクラスタ C_p からある文書 d_q (= C_q)を削除したときの avg_sim は，以下の式により求められる．

$$avg\_sim(C_r, C_r) = (cr\_sim(C_p, C_p) - 2cr\_sim(C_p, C_q) - ss(C_p) + 2sim(d_q, d_q)) / ((|C_p|-1)(|C_p|-2)) \quad (19)$$

図2の繰り返し処理のステップ1の a)では，ある文書をクラスタからいったん削除して別のクラスタに移した場合の G の値の変化を求める必要がある．上記(18), (19)式を用いることでこの計算が効率よく行えることになる．

3.5 インクリメンタルな更新

本研究では，新規文書が到着したとき，前回のクラスタリング結果を再利用することを想定している．具体的には以下のような処理を行うことになる．

1. 前回クラスタリング対象となった文書のうち，十分古くなった文書については，クラスタ中から削除する（削除の基準については[3,5]を参照）．
2. クラスタ代表および G の値を再計算する．
3. 新規に到着した文書も含め，図2の K-means 法の繰り返し処理を実行する．

継続的にニュース記事などが配信されるような環境においては，前回のクラスタリング対象の文書に新たな文書を追加しても，クラスタリング結果が大幅に異ならないことが予想される．そのため，前回の結果を次回のクラスタリングの初期配置として用いることは妥当な選択だと考えられる．

4. まとめ

以上，忘却の概念に基づくクラスタリング手法の改良手法について述べた．今後は実験に基づく提案手法の評価を行う予定である．

[謝辞]

研究の一部は，日本学術振興会科学研究費若手研究(B)(14780316)，基盤研究(B)(12480067)，および文部科学省科学研究費特定領域研究(14019009)による．

[文献]

[1] Yang, Y., Carbonell, J.G., Brown, R.G., Pierce, T., Archibald, B.T., Liu, X.: “Learning Approaches for Detecting and Tracking News Events”, *IEEE Intelligent Systems*, Vol. 14, No. 4 (1999).

[2] Allan, J. (ed.): *Topic Detection and Tracking: Event-based Information Organization*, Kluwer, 2002.

[3] Ishikawa, Y., Chen, Y., H. Kitagawa: “An On-Line Document Clustering Method Based on Forgetting Factors”, *Proceedings of 5th European Conference on Digital Libraries (ECDL 2001)*, pp. 325-339 (2001).

[4] Can, F.: “Incremental Clustering for Dynamic Information Processing”, *ACM TOIS*, Vol. 11, No. 2, pp. 143-164 (1993)

[5] 石川佳治，北川博之：忘却の概念に基づくインクリメンタルな文書クラスタリング手法，情報処理学会研究報告，Vol. 2001, No. 71 pp. 313-320 (2001)

[6] Cutting, D., Karger, D.R., Pedersen, J.O., Tukey, J.W: “Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections”, *Proc. ACM SIGIR*, pp. 318-329 (1992)

石川 佳治 Yoshiharu ISHIKAWA

筑波大学電子・情報工学系助教授．1994年筑波大学大学院博士課程工学研究科単位取得退学．博士（工学）．空間データベース，文書データベース，情報検索などに興味を持つ．ACM, IEEE-CS，情報処理学会，電子情報通信学会，日本データベース学会，各会員．

北川 博之 Hiroyuki KITAGAWA

筑波大学電子・情報工学系教授．1980年東京大学大学院理学系研究科修了．理学博士（東京大学）．異種情報源統合，文書データベース，WWWの高度利用等の研究に従事．著者「データベースシステム」(昭晃堂)，「Unnormalized Relational Data Model」(共著，Springer-Verlag)等．ACM，IEEE-CS，日本データベース学会，情報処理学会，電子情報通信学会，日本ソフトウェア科学会，各会員．