

オンラインニュースを対象とした モニタリングシステムの提案

Proposal of Monitoring System for Online News

沼野 航希* 高間 康史

Koki Numano, Yasufumi Takama

首都大学東京大学院システムデザイン研究科
Graduate School of System Design, Tokyo Metropolitan University

Abstract: 本稿では、オンラインニュースの定期的なモニタリングを支援する情報可視化システムを提案する。オンラインニュースは主要な情報源の一つとなっているが、新着記事が絶え間なく到着し、常時モニタリングすることは困難である。興味ある話題を見逃すことなく効率的にモニタリングするために、提案システムでは以前関心を抱いた話題の続報提示、モニタリングするタイミングを判断する手がかりの提示によりモニタリングを支援する。

1. はじめに

本稿は、オンラインニュースの定期的なモニタリングを支援する情報可視化システムを提案する。近年、オンラインニュースは Web 上で、主要な情報源の一つとなっている。ニュースサイトは多数存在し、2014 年 10 月 2 日の新着記事数は、「朝日新聞デジタル¹」が 129 件、「日本経済新聞²」が 270 件、「毎日新聞³」が 112 件であった。このように、ニュースサイト一つあたりの新着記事数は 100 件を超えるが、複数のニュースサイトを閲覧することが一般的であるため、一人のユーザが一日に受け取る新着記事は数百件になることも珍しくない。ユーザは日常生活において、これらの記事を継続的に全てモニタリングすることは困難であるため、モニタリングしていない間の情報の見逃しが発生することが問題としてあげられる。そのため、ユーザが関心を抱いている話題を効率的にモニタリングできるようにすることが重要と考える。

効率的なニュース閲覧を支援するサービスとして、ニュースキュレーションサービスが急速に普及しつつある。代表的なニュースキュレーションサービス

の一つである「グノシー⁴」は、独自のアルゴリズムでユーザの興味に合った最新ニュースを提示する他、時間指定によるプッシュ通知お知らせなどの機能があるが、適切なタイミングでニュースを確認できているか否かは考慮されていない。

本稿で提案するシステムは、話題検出・追跡技術を用いてユーザが関心を抱いている話題を可視化して提示する。また、ユーザがモニタリングするタイミングを判断する手がかりも可視化して提示する。本稿では、構築したプロトタイプシステムについて述べるとともに、ユーザに利用してもらった予備実験の結果について報告する。

2. 関連研究

2.1 ニュースキュレーションサービス

ニュースキュレーションサービスとは、Web 上のニュースを収集、分類を行いユーザに提供するサービスのことである。スマートフォンの普及に伴い、急速に利用者が増大している。代表的なキュレーションサービスに前述のグノシー⁴の他、SmartNews⁵などが挙げられる。

SmartNews は、エンタメ、スポーツ、グルメなど 11 のジャンルのの中から、読みたい話題を自由に選択

¹ <http://www.asahi.com/>

² <http://www.nikkei.com/>

³ <http://mainichi.jp/>

*連絡先: 首都大学東京大学院システムデザイン研究科
〒191-0065 東京都日野市旭が丘 6-6
E-mail: ytakama@sd.tmu.ac.jp

⁴ <http://gunosy.com/>

⁵ <http://www.smartnews.com/ja/>

し、並び替えることができる。また、Twitter でツイートされた Web ページをリアルタイムで解析し、話題になっている記事を配信する機能も備えている。

2.2 話題検出・話題追跡

時系列に到着する一連のニュースなどから新規に出現した話題を抽出することを話題検出、既出話題の続報を検出することを話題追跡と呼ぶ。テキストデータを対象とした話題検出・追跡の手法は様々な提案されている[2][3][4][5][6][7][8][9][10]。ニュース記事のようなテキストデータを対象とする場合、記事間または、記事と記事クラスタ間の類似度を求めることにより話題の抽出をする手法が一般的である。話題検出・追跡処理の一般的な流れを以下に示す。

1. 特徴量の計算…各記事から特徴ベクトルの生成
2. 文書クラスタリング…話題に対応したクラスタの生成

ステップ1では、クラスタリングを行う前処理として特徴量の計算を行う。記事及びクラスタの表現は、ベクトル空間モデルがよく用いられる[2][3][4][5]。ベクトル空間モデルでは、単語の重みは tfidf で求めることが多いが、上嶋ら[5]は、idf 値を更新することは、一度決定した過去のクラスタリング基準が変わってしまう場合があるという理由から tf 値のみを用いて逐次クラスタリングを行っている。菊池ら[2]は、過去の文書から事前に求めた idf 値を用いている。

ステップ2で行う文書クラスタリングの手法も様々な提案されている。クラスタの重心ベクトルと文書ベクトルの類似度を余弦尺度を用いて計算し、逐次クラスタリングにより話題クラスタを抽出する手法[2][3][4][5][6]、共起語集合が話題を形成するとの考えに基づき、共起語集合間の類似度 JS divergence を用いて計算し話題を抽出する手法[7]などが提案されている。JS divergence とは、2つの分布の相違度を測る尺度である KL divergence を対称化したものである。0 から 1 までの値をとり、値が大きいほど2つの分布は異なっている。

芹澤ら[4]は、コサイン類似度を用いて各トピック間の類似度を求め、連続する2日間の類似度が閾値以上ならばトピック間に関連付けを行うことでトピックを追跡する。

2.3 可視化表示システム

文書クラスタリングによって生成された話題クラ

スタをわかりやすくユーザに提示するために、時系列ごとに話題の遷移を示すインタフェースも様々な提案されている[3][6]。

森ら[3]は、2次元平面上の横軸に時間軸を、話題クラスタを縦軸に配置して話題遷移を可視化する手法を提案している。話題の分岐、収束の両方を確認することができ、前後関係や話題の追跡が容易になるとしている。

長谷川ら[6]が提案する T-Scroll は、時系列文書を対象とするクラスタリングシステムが定期的に生成するクラスタリング結果をもとに、クラスタ間の関連を巻物状に可視化する。楕円でクラスタを示し、その中には、そのクラスタを最も適切に表すようなキーワードを選んで表示する。また、楕円のマウスオーバー時に、クラスタに含まれる文書一覧を表示する機能も備えている。楕円どうしを繋ぐことにより、話題の時系列変化を把握し、クラスタの内容を容易に確認できることがこのシステムの特徴である。

3. オンラインニュースを対象としたモニタリングシステム

本稿では、オンラインニュースの定期的なモニタリングを支援する情報可視化システムを提案する。具体的には、前回モニタリング以降に到着したオンラインニュースについて、新規に発生した話題に関する記事、前回関心を持った話題の続報記事の発見を支援する。提案システムは、オンラインニュースの収集、文書クラスタリングによる話題検出・追跡、インタフェースによる提示から構成される。図1にシステム構成図を示す。以下では構成要素それぞれについて説明する。

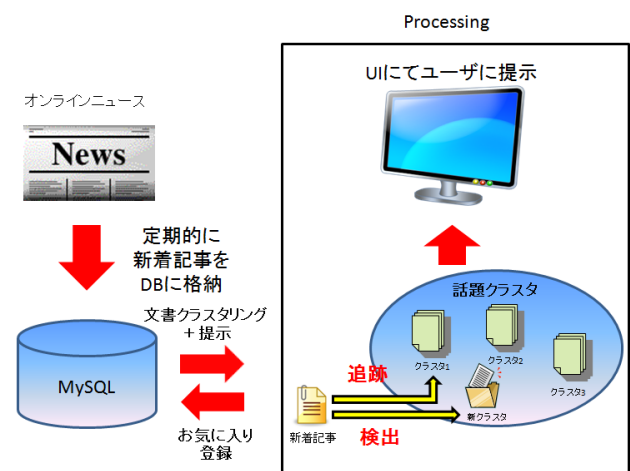


図1. システム構成図

3.1 オンラインニュースの収集

モニタリングシステム構築にあたり、オンラインニュースを Web 上から収集する。オンラインニュースに含まれる情報は title, date (配信日時), text (記事本文) であり、これらの情報を得るために、Ruby の `rubygems` ライブラリである `Mechanize` を使用する。4 節で述べる実験では、朝日新聞デジタル⁶の新着記事を 2014 年 6 月 1 日～6 月 30 日の期間取得して、記事ごとにデータベースに格納したものをを用いている。表 1～表 3 にデータベースの構成について示す。newstable (表 1) は、記事内容と記事の配信日時を格納するためのテーブルである。apclustertable (表 2) は、話題クラスタ毎の記事番号を格納するためのテーブルである。favoritefeednotable (表 3) は、ユーザが記事および話題クラスタをお気に入り登録した際に、記事番号、配信日時、お気に入り登録された回数を格納するためのテーブルである。インタフェースで提示する際に、関心のある話題クラスタに関して配信日時の情報を必要とするため、favoritefeednotable にも date を格納する。

表 1. newstable

カラム	内容
id	記事番号 (1 から順に auto_increment)
title	記事のタイトル
date	記事の配信日時
text	記事本文

表 2. apclustertable

カラム	内容
clusterno	クラスタナンバー (1 から順に auto_increment)
feedno	記事番号 (newstable の id)

表 3. favoritefeednotable

カラム	内容
id	記事番号 (newstable の id)
date	記事の配信日 (newstable の date)
count	お気に入り登録した回数

3.2 文書クラスタリング

文書クラスタリングには Affinity Propagation[1] アルゴリズムを用いる。Affinity Propagation アルゴリ

ズムは、予めクラスタ数を決めておく必要がなく、クラスタリング結果が初期値に依存しないという特徴を持っている。本稿で対象とするオンラインニュースは時系列的に発生するため、予めクラスタ数を決めることができないことから、クラスタリング手法に Affinity Propagation アルゴリズムを用いた。Affinity Propagation アルゴリズムは、全要素間の関係性を similarity として設定し、availability と responsibility という 2 種類のメッセージを交換し合うことで、exemplar (クラスタの中心) を決定し、クラスタを生成する手法である。responsibility($r(i,j)$), availability($a(i,j)$) は共にデータポイントが j が i の exemplar としてふさわしい度合いを表すが、前者は i が j を選ぶ度合いであり、 i から j へ送られるのに対し、後者は j が i にとってふさわしい度合いであり、 j から i に送信される。Affinity Propagation アルゴリズムのフローチャートを図 2 に示す。

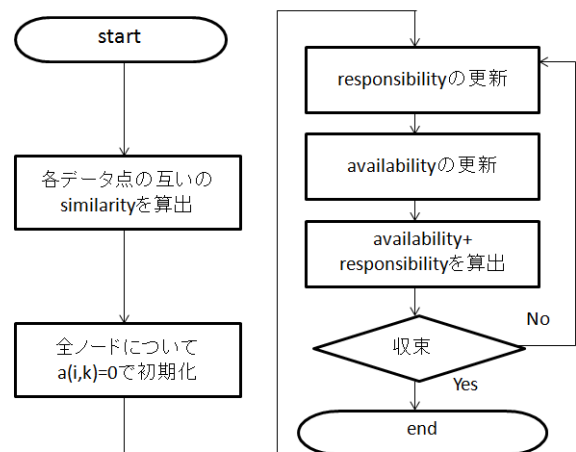


図 2. Affinity Propagation アルゴリズムのフローチャート

本稿では、similarity の算出に cos 類似度を用いる。similarity 算出の手順を示す。

1. newstable の title と text を形態素解析し、名詞と未知語を抽出
2. 抽出した単語を特徴とし、tfidf 値を重みとして、各新聞記事の特徴ベクトルを生成
3. 全記事対の cos 類似度を計算

ステップ 3 の結果に基づき、newstable に格納されている記事数を N として $N \times N$ の類似度行列を生成し、Affinity Propagation アルゴリズムに入力する。

Affinity Propagation アルゴリズムの収束条件は、クラスタ割り当てが直前の結果と変化がない場合、

⁶ <http://www.asahi.com/>

または計算の反復回数が最大値を超える場合である。今回は予備実験の結果に基づき、反復回数を 50 回としてクラスタリングを行った。

availability および responsibility は以下の式で算出される[11]。

$$r(i, j) = (1 - \lambda) * \rho(i, j) + \lambda * r(i, j) \quad (1)$$

$$a(i, j) = (1 - \lambda) * \alpha(i, j) + \lambda * a(i, j) \quad (2)$$

ここで、 λ は Damping Factor と呼ばれる、反復計算の中で availability および responsibility が振動するのを防ぐための係数である。今回は、予備実験の結果に基づき $\lambda=0.9$ で反復計算を行った。また、 $\rho(i, j)$ と $\alpha(i, j)$ は以下の式から計算する。

$$\rho(i, j) = \begin{cases} s(i, j) - \max_{k \neq j} \{a(i, k) + s(i, k)\} & (i \neq j) \\ s(i, j) - \max_{k \neq j} \{s(i, k)\} & (i = j) \end{cases} \quad (3)$$

$$\alpha(i, j) = \begin{cases} \min \{0, r(j, j) + \sum_{k \neq i, j} \max \{0, r(k, j)\}\} & (i \neq j) \\ \sum_{k \neq i} \max \{0, r(k, j)\} & (i = j) \end{cases} \quad (4)$$

3.3 インタフェース

3.2 節に示した文書クラスタリング結果に基づき、話題クラスタをユーザに提示するためのインタフェースを提案する。開発には Processing を用いた。MySQL Server から新聞記事及び話題クラスタに関する情報を取得し、Processing にて提示する。提案するインタフェースには、続報記事数提示モード、リストモードの二つのモードがある。各モードのスクリーンショットを図 3、4 にそれぞれ示す。

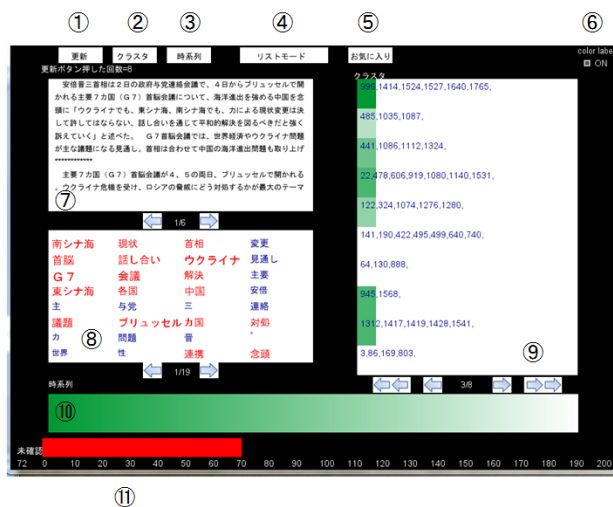


図 3. インタフェース（リストモード）

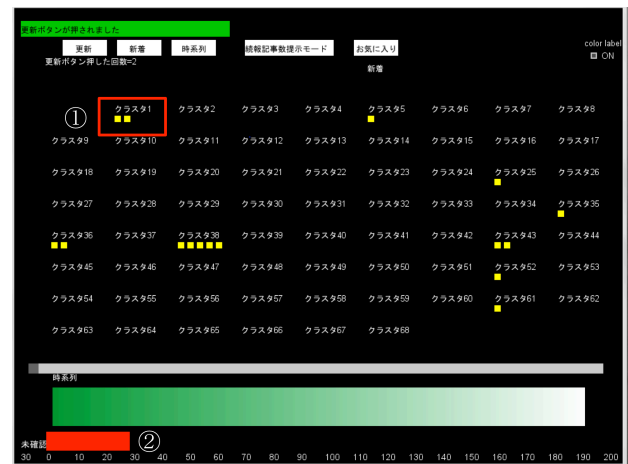


図 4. インタフェース（続報記事数モード）

ユーザはこのインタフェースを使用して、新着記事及び話題クラスタ毎の記事内容の確認や、関心のある話題クラスタの続報記事数の確認を行う。

リストモード（図 3）では、新着記事及び話題クラスタ毎の記事内容の確認を行い、続報記事数提示モード（図 4）では、関心のある話題クラスタの続報記事数の確認を行う。それぞれのモードについて、以下に機能の説明を示す。

リストモード（図 3）では、新着記事の見出し、または話題クラスタ毎に記事番号を確認できる（図 3 の⑨）。新着記事⇄話題クラスタの切替は、新着⇄クラスタ切替ボタン（図 3 の②）で行う。リスト（図 3 の⑨）内のテキストをクリックすると、新着記事の場合は記事内容が表示され（図 3 の⑦）、話題クラスタの場合は、クラスタ内の記事内容（図 3 の⑦）とクラスタ内の単語についてのタグクラウドが表示される（図 3 の⑧）。新着記事及び話題クラスタリストで関心を抱いたものがあれば、お気に入りボタン（図 3 の⑤）により、お気に入り登録を行う。

また、リスト内の着色の方法に関して、時系列⇄関心の有無切替ボタン（図 3 の③）により、時系列または関心の有無のどちらを基準に着色するのかわりを切り替えることができる。新着記事⇄話題クラスタ切替ボタンとの組み合わせにより、以下の 3 機能が利用可能である。ここで、新聞記事と時系列は同種の情報であるためその組み合わせは除外している。

機能 1. 新着+関心の有無

→新着（未確認）記事の中で、前回関心を持っていた記事の続報を強調

機能 2. クラスタ+時系列

→各クラスタの最新記事を比較：より最新の記事を含むクラスタを強調

機能 3. クラスタ+関心の有無

→前回お気に入り登録した記事が多く含まれるクラスタを強調

続報記事数提示モード（図 4）では、話題クラスタ毎に、ユーザがまだ内容を確認していない続報記事数を黄色い四角で提示する（図 4 の①）。なお、続報記事数が提示されるクラスタは、ユーザが前回お気に入り登録をした記事が含まれるクラスタのみである。また、ユーザが前回記事内容を確認してから到着した新着記事数を赤いバーとともに表示する（図 4 の②）。

リストモードと続報記事数提示モードの切替は、リストモード⇄続報記事数モード切替ボタン（図 3 の④）により行う。

4. 評価実験

4.1 実験概要

本実験では、20 代の工学系大学院生を対象に、提案したシステムのプロトタイプを使用してモニタリングを行ってもらった。本実験の検証目的は、短時間で関心のある記事を確認できること、記事内容を確認する必要があるか否かを判断できることの 2 つである。3.1 節で述べた通り、実験には、朝日新聞デジタルの新着記事⁷を 2014 年 6 月 1 日～6 月 30 日の期間取得して用いた。モニタリングは本務の合間に行われるとの想定に基づき、実験協力者には他の作業を適宜してもらいながら、提案システムを利用してもらった。以下に実験手順を示す。

1. 更新ボタンを 1 回押して新着記事あるいは新規クラスタから 5 個ずつ計 10 個お気に入り登録
2. 実験開始時間から 5 時間の間に、ユーザの任意のタイミングで続報記事数を確認（回数は 13～15 回）
3. 記事内容の確認が必要か否かを判断
必要と判断した場合→4-(a)へ
必要でないと判断した場合→4-(b)へ
- 4-(a). 5 分の制限時間で記事内容を確認し、適宜お気に入り登録
- 4-(b). 他の作業の再開
5. Step2～Step4 をユーザが制限時間内に繰り返し試行

実験において、ステップ 3-(a)の記事内容確認は 5 回に限定した。これにより、新着記事があまりない

状態で確認してしまうと、後の方で多数の記事を一度に確認しなくてはならない状況が発生することになる。従って、5 分間で確認できる程度の新着記事が到着した、適切なタイミングを実験協力者が判断可能かどうかを検証可能と考える。

4.2 実験結果

表 4 に、各実験協力者の記事内容確認時間を示す。表 5 に、関心のある話題に関して追跡ができた割合として、続報記事を含む話題クラスタを確認した割合（左セル）、続報記事を確認した割合（右セル）を示す。表 6 に、前回の確認時から到着した記事数（左セル）、続報記事数（右セル）を示す。

表 4. 記事内容確認時間

	A	B	C	D
1 回目	4 分 01 秒	2 分 42 秒	5 分 44 秒	5 分 52 秒
2 回目	3 分 43 秒	3 分 02 秒	9 分 59 秒	7 分 00 秒
3 回目	2 分 38 秒	3 分 25 秒	7 分 13 秒	7 分 45 秒
4 回目	1 分 53 秒	4 分 51 秒	7 分 16 分	8 分 32 秒
5 回目	1 分 41 秒	3 分 52 秒	5 分 49 秒	4 分 01 秒

表 5. 話題追跡できた記事数の割合
(左：話題クラスタ確認、右：続報記事確認)

	A		B		C		D	
1 回目	0.67	0.40	0	0.57	0.33	0.78	1.00	0.22
2 回目	1.00	0.50	0.71	0.10	1.00	0.92	1.00	0.79
3 回目	0.10	0.10	1.00	0.23	1.00	0.97	1.00	0.93
4 回目	0	0.08	1.00	0.32	1.00	1.00	0.93	1.00
5 回目	0.18	0.08	1.00	0.22	1.00	0.88	0.70	0.86

表 6. 未確認の新着記事到着件数
(左：到着記事数、右：続報記事数)

	A		B		C		D	
1 回目	60	5	80	7	80	9	111	18
2 回目	51	10	81	10	132	13	50	14
3 回目	160	30	72	13	101	33	72	14
4 回目	100	13	138	31	96	28	212	43
5 回目	74	13	74	27	118	60	40	14

実験結果より、実験協力者 A, B は全ての回に 5 分以内で記事内容を確認しているのに対し、C, D はほとんどが 5 分を超えていることがわかる。C, D は、到着した続報記事、続報を含む話題クラスタを高い割合で確認していることが、記事確認に多くの時間を要した原因であると考えられる。

適切なタイミングでモニタリングを行えているか

⁷ <http://www.asahi.com/>

否かに関して、実験協力者毎の記事内容確認時間のばらつきについて考察する。1 回目はモニタリング開始のため2 回目以降よりも時間がかかること、および5 回目は実験終了の制約があることを考慮して、2~4 回目のモニタリングにおける確認時間の最大値、最小値の差を見ると、実験協力者 A は1 分 50 秒、B は1 分 49 秒、C は2 分 46 秒、D は1 分 32 秒であった。実験協力者 C は確認時間が長いことを考慮すれば、実験協力者によらず確認時間のばらつきは大きくなく、適切なタイミングで確認が行えていると考える。また、実験協力者 A 以外は、1 回目を除き 70%以上関心のある話題クラスタあるいは続報記事を確認できていることがわかる。1 回目は、インタフェースに関して操作の要領を得ていないことが原因として考えられる。A は、関心を持った話題クラスタあるいは新着記事を効率よく見つけられなかった可能性がある他、それらの話題に対して次の確認時興味を失い、確認をしなかった可能性もあるため、今後調査の必要があると考える。

実験後、話題検出・追跡、提示するタイミング、システム全体のことに関して、アンケートを実施した。話題検出・追跡に関しては、ほとんどが続報を確認できたと回答しているが、「サッカー」という話題でも、自分の関心の無い国に関する話題も続報としてまとめられていたため、全ての続報に関心を持ったとは言えないという意見があった。データ収集時期がワールドカップ開催期間であったため、サッカーに関する話題として広い範囲で一つの話題クラスタが生成されてしまったことが原因であると考ええる。この問題を解決するには、より時系列を考慮したクラスタリング手法が必要と考える。提示するタイミングに関しては、ほとんどが適切なタイミングで記事内容を確認できたと回答している。システム全体に関しては、画面の遷移がわかりにくいことや、どの記事がどの記事の続報であるかわかりにくいなどの意見があった。話題追跡に関して、A 以外は高い割合でできていることから、インタフェースの完成度を上げることで、より短時間で記事内容を確認できることが期待できる。

5. おわりに

本稿では、オンラインニュースの定期的なモニタリングを支援する情報可視化システムを提案した。提案するシステムでは、ユーザが関心を抱く話題の追跡だけでなく、話題クラスタごとに続報記事数を提示することで、より適切なタイミングで記事内容を確認することを支援する。プロトタイプシステムを構築し、評価実験を行った結果、高い割合で関心

のある話題クラスタについて追跡できることを示した。

今後は、時系列を考慮したクラスタリング手法に改善するとともに、インタフェースの操作性を向上し完成度を高めることで、より見やすいインタフェースを検討する予定である。

参考文献

- [1]B. J. Frey and D. Dueck: Clustering by passing messages between data point, Science, Vol. 315, pp. 972-976, 2007.
- [2]菊池匡晃, 岡本昌之, 山崎智弘: 階層型クラスタリングを用いた時系列テキスト集合からの話題抽出, DBSJ Journal, Vol. 7, No. 1, pp. 86-90, 2008.
- [3]森幹彦: ニュース記事の話題分岐を時系列で追跡可能な可視化法, 情報処理学会第 71 回全国大会, 6B-3, 2009.
- [4]芹澤翠, 小林一郎: 潜在トピックの類似度に基づくトピック追跡への取り組み, 第 25 回人工知能学会全国大会, 3F3-2, 2011.
- [5]上嶋宏, 三浦孝夫, 塩谷勇: 時系列ニュース記事集合に基づくニュース記事の順序付け, DEWS2004, 1-B-04, 2004.
- [6]長谷川幹根, 石川佳治: T-Scroll: 時系列文書のクラスタリングに基づくトレンド可視化システム, 情報処理学会論文誌: データベース, Vol. 48, No. SIG 20 (TOD 36), pp. 61-78, 2007.
- [7]森井洸明, アダムヤフト, 田中克己: ニュースアーカイブを用いた話題変化と原因語の発見, DEIM Forum 2012, D4-2, 2012.
- [8]橋本泰一, 村上浩司, 乾孝司, 内海和夫, 石川正道: 文書クラスタリングによるトピック抽出および課題発見, 社会技術研究論文集, Vol. 5, pp. 216-226, 2008.
- [9]高橋祐介, 横本大輔, 宇津呂武仁, 吉岡真治, 河田容英, 神門典子, 福原知宏, 中川裕志, 清田陽司: 時系列トピックモデルにおけるバーストの固定, DEIM Forum 2012, F5-5, 2012.
- [10]平田紀史, 大園忠親, 新谷虎松: ユーザの選好に基づくトピック分析システムの試作, JSAI2008, pp. 277-277, 2008.
- [11]藤原靖宏, 入江豪, 北原友恵: Affinity Propagation のための高速化手法, DEIM Forum, C1-3, 2012.