203-14in

バースト依存型リランキングによるトレンド表現文の推定

Estimate of the trendy expression sentence by burst-dependent model reranking

難波 悟史　但馬 康宏　菊井 玄一郎
Satoshi Namba　Yasuhiro Tajima　Genichiro Kikui

岡山県立大学大学院情報系研究科
Graduate School of Systems Engineering，Okayama Prefectural University

This paper proposes a method for extracting a micro-blog article that most explains a search-query burst. The proposed method consists of 2 steps. The first step locates the strongest burst period in the stream of micro-blog articles that contain the given burst-query word. The second step chooses the article with the highest tf-ridf (residual inverse document frequency) from all the articles posted within the detected burst period. Experimental results show the proposed method successfully extracted correct articles for approximately 77% of burst queries.

1. はじめに

流行や話題に挙がっている事柄をここでは「トレンド」と呼ぶ．このトレンドをうまく理解することができるなら，それは世の中で注目されている内容を効率よく理解できることにつながる．インターネット上のトレンドを知る手掛かりの一つとして，「検索急上昇ワード」と呼ばれる単語が挙げられる．検索急上昇ワードとは，ある期間に検索サイトに入力される検索語（単語あるいは複数単語からなる文字列）のうち，それより前の期間の入力数を比べて急増したものをいう．しかし，急上昇ワードだけでは，その単語が何であるか，また，何が起きて検索数が急激に増えたのか分かるとは限らない．例えば「グリッドロック」のような，あまり聞きなれないような単語の場合，多くの人はそれが何を意味するのか分からない．また，もし仮にこの単語が渋滞現象を表すと知っていたとしてもなぜ急上昇ワードになったのか分からないかも知れない．そのような場合，ユーザは自分でその単語をウエブ検索してウエブ検索をして，情報を読んで理解する必要があり煩雑である．

この問題に対して，急上昇ワードの「急上昇要因」を表すキーワードをマイクロブログから自動抽出するという研究が行われている[菊井 12]．この研究では検索急上昇ワードを含むテキストの投稿件数がマイクロブログにおいて検索クエリーと同時あるいは先行的に急上昇することに注目し，マイクロブログにおけるこれらの投稿から高頻度のキーワードを抽出することにより，上位3位以内に 72%の精度で急上昇要因を示すキーワードの抽出を実現している．しかしながら，キーワードの提示では要因を表す事象の理解が容易とは言えない．

そこで，本研究では上記研究を発展させ，検索急上昇ワードを入力すると，その急上昇ワードを特徴づけられるようなトレンドがどういうものかを表現するような文（例えばグリッドロックの例では「＜グリッドロック＞「超」渋滞現象、震災で初確認（毎日新聞） - Y!ニュース」といった文）を極力早期に抽出する手法を提案する．

以下，2 章では提案手法の考え方，3 章では提案手法，4 章では提案手法の評価実験と考察について述べる．

2. 基本的な考え方

2.1 抽出対象文書

トレンドを表現する文章を早期に抽出するためには，抽出元のドキュメントがトレンドに敏感であることが必要である．従って，ブログやマイクロブログのようにユーザが日々書き込むメディアが候補となる．

[菊井 12]によると，検索クエリーログ，ブログ，マイクロブログ(twitter)について，それぞれ分析対象の検索急上昇ワードを含むテキスト数（クエリーログにおいてはクエリ数）の時間変化を見たとき，マイクロブログにおいて一番早くバースト(単位時間あたりの投稿数が通常と比べて大きく増加している時間帯)が検出される傾向が見られた．このことから twitter は blog 記事よりトレンドに敏感であると考えられる．

そこで，本研究では twitter を対象としてトレンドを表現する文章の抽出を行うことにした．

2.2 バースト要約としてのトレンド表現文抽出

先に述べたとおり，先行研究[菊井 12]によれば検索急上昇ワードの多くはマイクロブログにおいてもそれを言及する投稿記事数が急増(バースト)する．また，これらの記事の集合において出現頻度上位の単語には急上昇を説明するキーワードが 70%以上の割合で含まれていることが示されている．従って，バースト記事の集合を代表するような記事を特定することでトレンドを表す文章を出力できると考えられる．

本研究では記事集合を代表する記事を選ぶために，文章要約における文の重要性をランキングする手法を用いる．

次章で手法の詳細について説明する．

3. 提案手法

3.1 提案手法全体の流れ

提案手法は急上昇ワードを入力としてそのトレンドを説明する文章を出力する．手法全体の流れを図 1 に示す．

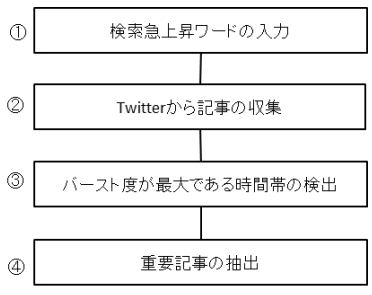

図 1 提案手法の流れ

連絡先：岡山県立大学情報工学部，岡山県総社市窪木 111，
E-mail:kikui’at’cse.oka-pu.ac.jp

まず，①で検索急上昇ワード (一つの単語あるいは複数単語の連鎖)を入力として受け付ける．②では，Twitter の searchAPI を用いて①で与えられた検索急上昇ワードを含み，この急上昇ワードがポータブルサイトで発表されるより前の記事を収集する．収集した記事の集合を「(当該急上昇ワードに対する)Tweet セット」と呼ぶことにする．次に③では，収集した Tweet セットに対してバーストを検出する．もしバーストが複数検出された場合，後述するバースト度が最大のバーストの時間帯を検出する．最後に④で，③によって検出された時間帯に投稿された各記事に対して，重要度を表す値を与える．この重要度をもとに記事を需要度の降順に並び替え，ランキング上位の記事を提示する．以上が一連の流れとなる．なお③でバースト検出できなかったときは②で収集できた記事全体を用いて重要度を算出する．

3.2 バースト検出

提案する手法ではバーストの検出に Kleinberg の提案する検出手法[Kleinberg 02]を用いた．この手法ではドキュメントストリーム(twitter や電子掲示板などの時間情報がある文書の集合)中の各ドキュメントに対して，通常状態とバースト状態の 2 つの状態を考え，これらをオートマトンでモデル化する．状態および状態遷移に対してコストが定義されており，最小コストとなる状態列においてバースト状態になっている文書の列をバースト期間として固定する．この手法はパラメータの設定によってバーストとして捉えるべき投稿時間の間隔や状態遷移のコストなどをコントロールできるが，多様なドキュメントストリームに対して適切なバースト検出を行うように調整することは必ずしも容易でない．そこで，ドキュメントストリームを一定の時間ごとに区切り，その時間帯に投稿されたドキュメントに対して投稿時間の間隔が均等になるように，ドキュメントの投稿時間を振りなおし，バースト判定を行うようした．

バーストが複数検出された場合には「バースト度」が最大のバーストを選ぶ．ここで「バースト度」とは当該バースト期間における各ドキュメントに対してバースト状態とした場合のコストからバースドでない状態と仮定したコストの差を取ったものの総和と定義される [藤木 04]．形式的には当該バースト期間において下記の$C_B(\mathrm{t})$の総和を取ったものである．

$$C_B(\mathrm{t}) = C_1(t) - C_0(t) \quad \cdots\cdots (1)$$

$$C_1(t) = -\ln\left(\frac{sK}{T}e^{-\frac{sK}{T}x_t}\right) + \min_l(C_l(t-1) + \tau(l, j)) \quad \cdots\cdots (2)$$

$$C_0(t) = -\ln\left(\frac{K}{T}e^{-\frac{K}{T}x_t}\right) + \min_l(C_l(t-1) + \tau(l, j)) \quad \cdots\cdots (3)$$

ここで，ドキュメント番号をt，t番目のドキュメントd_tとする． $C_B(\mathrm{t})$はバースト度を示し，$C_1(t)$ はd_tのバースト状態のコスト，$C_0(t)$は平常状態のコストを表す．K はドキュメントストリーム中に含まれるドキュメントの数，T はドキュメントストリーム全体の時間を表し，x_tはd_tとd_{t-1}の到着時刻の差を表す．S は任意のパラメータとし，$\tau(l, j)$は状態遷移コストを表す．lはd_{t-1}の状態を，jはd_tの状態をそれぞれ表す．$\tau(l, j)$は$l < j$ならば $\tau(l, j)$=γ(任意のパラメータ)，それ以外の場合は $\tau(l, j)$=0 である．

3.3 重要度付与

文の重要度の計算においては，テキスト自動要約[奥村 05]などで用いられる重要文抽出の手法を用いる．今回は重要文抽出の中でも代表的な一つの「テキスト中の単語の重要度を利用する」方法を用いる．重要な単語を多く含む文ほど重要であるという考え方である．具体的にはある記事 A の重要度 W(A)をその記事に含まれる各名詞列に重要度の総和と考え，次の式(4)のように定義する

$$\mathrm{W(A)} = \sum_{k \in D} S(k) \quad \cdots\cdots (4)$$

ここで，D は記事 A における名詞列の集合，S(k)は名詞列 k に対するスコアである．また，S(k)の計算式として， 情報検索のインデックス語抽出で用いられる$tf\text{-}ridf$値を用いる．

$tf\text{-}ridf$値はtf値[徳永 99]と$ridf$値[北 02]を掛け合わせたものである．ドキュメント集合中の全テキスト数を N，単語tが出現するテキストの数$df(t)$を用いて，以下のように表すことができる．

$$tf(t, D) = \text{文書 D 中の単語}t\text{の出現回数} \quad \cdots\cdots (5)$$

$$ridf = log\frac{N}{df(t)} - log\frac{N}{N \cdot \left(1 - e^{-\frac{df(t)}{N}}\right)} \quad \cdots\cdots (6)$$

tf値を計算するための文書 D は要約対象の文書集合，すなわち，バースト期間中の全 tweet(を1文書とみなしたもの)を用いた．

また，$ridf$値におけるドキュメント総数 N は twitter 社で公開している「streaming API」を用いて public timelines から集めたデータ(全公開ツイートの 1%サンプリングデータ)における記事数，$df(t)$はこの tweet のうち，単語tを含む記事とした．バースト期間中の tweet 記事集合を使わなかったのはこれらの記事集合が偏りすぎているためである．

4. 提案手法の評価実験

提案手法がどのくらいの精度で検索急上昇ワードを含む文章を抽出ができるのか実験を行った．実験に用いた検索急上昇ワードは「Yahoo!検索ランキング*1」の「急上昇ワードランキング」で公表されている検索急上昇ワードである．これらを用いた理由は，検索急上昇ワードが急上昇となった理由(由来)が付与されていることである．また対象とした急上昇ワードは 2012 年 10 月 1 日~2012 年 10 月 5 日までに発表された 150 単語である．また，分析するドキュメントストリームは twitter から Twitter 社が公開している「search API」を用いて検索急上昇ワードを検索語として集めた記事である．実験当時は検索時間から最大 10 日まで遡れて記事を収集できたので，各検索急上昇ワードに対して，発表後 2 日後*2に searchAPI による記事の収集を行い，検索急上昇ワード発表より前に投稿された記事を評価の対象とした．なお，収集できた記事数が 0 件となった 5 つの検索急上昇ワードは評価対象外とした．図 2 はある検索急上昇ワードの投稿件数と時間変化の推移に検索急上昇ワード発表時刻と記事を収集した日を書き込んだものである．実線で囲んだ部分が評価の対象とした範囲である．

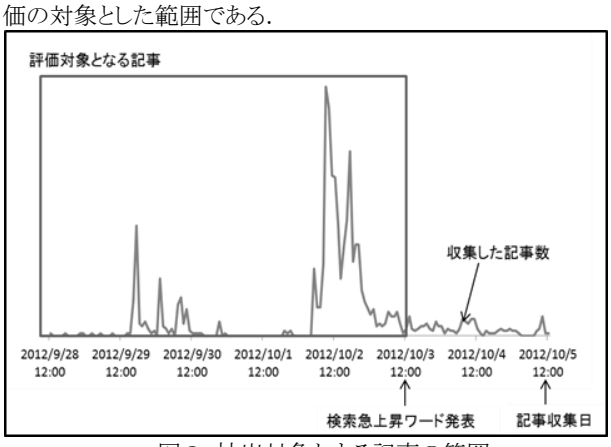

図 2 抽出対象とする記事の範囲

抽出した記事の正解の判断は，「Yahoo!検索ランキング」の「急上昇ワードランキング」に記載されている説明文(急上昇ワードになった理由)を基準とし，システムの出力したランキング上位

*1 http://searchranking. yahoo. co. jp/burst_ranking

*2 2 日後にしたのは取りこぼしを防ぐためでる

の記事がこれと同義であるか否か，人手で判断する．比較対象として，重要度計算に定数，tf値のみを用いたものについても精度を計算した．これらを図 3 に示す．

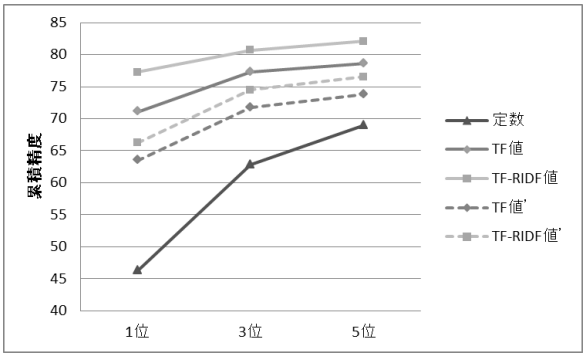

図 3 提案手法の結果

図 3 の横軸は記事の重要度の順位(降順)を表し，縦軸は各順位までに正解記事を抽出できた累積精度を表す．実線の TF-RIDF 値が提案手法の結果，実線の定数，TF 値は単語の重要度計算の方法のみを変えたものである．破線の TF 値‘，TF- RIDF 値’はバースト検出を行わずに，急上昇ワード発表以前に投稿された全記事を用いて，記事をランキングした結果である．

$tf\text{-}ridf$ 値を用いることで 77%の精度で，トレンドを表現する記事を提示することができた．また重要度が 5 位までものを出力することで 82%の精度で提示が可能であり，バースト検出を用いることで，約 10%の精度の向上が見られたことから，バースト検出が有効であることが確認することができた．

成功と失敗の例は次の通りである．ここで，下記の検索急上昇ワードとは重要文抽出を行った時の検索急上昇ワード，要因は Yahoo!検索ランキングに記載されていた検索急上昇ワードの要因，抽出記事は，重要文抽出をしたとき重要度が一番高かった記事を表す．まず抽出の成功例として，

・検索急上昇ワード 「グリッドロック」

・要因

「渋滞現象。東日本大震災時に東京都心で同時多発的に発生」

・抽出記事

『＜グリッドロック＞「超」渋滞現象、震災で初確認（毎日新聞） - Y!ニュース

東日本大震災が起きた昨年3月11日、車両が道路上に滞留してほとんど動かなくなる「グリッドロック」と呼ばれる渋滞現象が東京都心で同時多発的に起こっていたことが判明』

抽出に成功したものの多くは，成功例の記事のように記事の文字数が多いもの，すなわち，投稿記事数がもともと多いものであった．

次に抽出に失敗した例である．抽出に失敗した Tweet セットの多くは正解となる記事の文字数が他の記事に比べて文字数が少ないことが特徴として挙げられる．例として以下のようなものを挙げる．

・急上昇ワード 「ワーキングメモリー」

・要因

「作業を行うための短期記憶力。「ホンマでっか!?TV」SP で」

・抽出結果

『ワーキングメモリーが小さい＝IQ が低い＝バカ＝不幸＝離婚しやすく結婚しにくい。解決方法：直近のことに集中する。知識人に常に隣にいてもらい、随時分からないことをきく。人の話を深読みする。自分以外の他人を考えの中に登場させると忘れにくい。気になった事はメモ。』

また，このとき正解となるような記事は以下のようなものであった．

『ホンマでっか TV で、菅野美穂のお悩み、私も同じ。思いついた事をすぐ忘れる…ワーキングメモリーの容量が小さいのか (^_^;) なるほど』

このような失敗の原因として，用いた重要度計算式は記事の文字数が長くなるほど記事に含まれる単語が増えるため重要度が高くなってしまう．このために正解の記事より単語を多く含む記事が抽出されたのではないかと考えられる．

また，失敗した文字数が多い記事の中には現在のトレンドを紹介するような特徴的な記事を抽出していたものがあった． 以下がその例である．

・急上昇ワード 「片岡夕子」

・要因

「岐阜県各務原市で殺害され交際相手の男を殺人容疑で逮捕」

・抽出結果

『GoogleTrend 1:伴都美子 2:南乃”彩希 3:日本触媒 4:台風 5:運行情報 6:みなみのさき 7:hiroko 8:ビッグダディ 9:高橋秀人 10:片岡夕子 11:気象庁 12:シェアハウス 国際交流協会 13:警報 』

このようなトレンドを紹介するような記事はほとんどの Tweet セットのバースト期間内で多く見られたが，この失敗した例のほとんどが Tweet の記事数が少なかった．Tweet セットではトレンドを紹介する記事の数が多く，Tweet セット内このような記事を含む割合が大きくなってしまうので，$tf\text{-}ridf$値を用いている重要度計算式では，重要度が高くなってしまうのが原因だと考えられる．しかし，これらの記事は記事の内容が「1:単語 2:単語 3:単語 ・・・」のように特徴的であるため，記事の排除が可能であると考えられる．

最後に，失敗した検索急上昇ワードの中には，収集できた記事数が少なく，正解となる記事が Tweet セットに含まれていないものがあった．以下がその例である．

・急上昇ワード 「鈴木福が芸能界デビューした番組」

・要因

「ポイントサイトのクイズで出題」

一般的に検索エンジンに入力される検索語は，ユーザ自身の興味があるもの，調べたいものであるのに対し，twitter などに投稿される記事はユーザがある事柄に対する報告や感想，意見を投稿するものである．これらの違いにより，検索急上昇ワードであるのにかかわらず，twitter の記事数が少ない，または記事数が無いなどという状況が起きているのではないかと考えられる．また，今回は検索急上昇ワードの文字列をそのまま用いて記事を検索したため，複雑な(長い)急上昇ワードについては条件が厳しすぎて収集記事数が少なかったものもあった．収集記事数が少ないため，バースト検出が正しくできていないものも若干見られた．これらの問題を改善するためには，収集記事に含まれる単語を用いて再度記事を収集する(関連性フィードバック)，収集するときの検索語の条件を OR 条件にするなど条件を緩和することで，収集記事数を増やすことが考えられる．

5. おわりに

本稿では，ドキュメントストリームからバーストと検出された記事を用いることで，早期にトレンドを表現する文章を抽出する方法を提案し，一位正解確率 77%達成した．

今後の課題として，トレンドを紹介するような記事を排除すること，また，候補となる記事の収集カバー率を上げることなどが考えられる．

謝辞:
本研究の一部は科学研究費(24500296)の助成を受けたものである.

参考文献

[菊井 12] 菊井玄一郎，門内 健太，高橋 寛幸 :検索ホットワードとブログ系テキストの関係を探る， 第二回テキストマイニングシンポジウム, pp. 31-36(2012)

[Kleinberg 02] Jon Kleinberg: Bursty and hierarchical structure in streams, In Proc. The 8th ACM SIGKDD International Conference on knowledge Discovery and Data Mining (2002)

[藤木 04] 藤木 稔明， 南野 朋之， 鈴木 泰裕， 奥村 学: Document stream における burst の発見, 社団法人 情報処理学会 研究報告, 2004-NL-160, pp. 86-88 (2004)

[奥村 05] 奥村 学, 難波 英嗣:テキスト自動要約, オーム社, pp. 21-23（2005）

[徳永 99] 徳永健伸: 情報検索と言語処理, 東京大学出版会, pp. 26-27（1999）

[北 02] 北 研二, 津田 和彦, 獅々堀 正幹:情報検索アルゴリズム, 共立出版株式会社，pp. 43-44(2002)