# キーワードの自動拡張に基づくイベント言及ツイートの収集

五十嵐 祐貴 †* 大野 雅之 †* 岡崎 直観 ‡ 乾 健太郎 †
東北大学 † 科学技術振興機構さきがけ ‡
yuki.i@bc.tohoku.ac.jp {masayuki.ono, okazaki, inui}@ecei.tohoku.ac.jp

## 1 はじめに

近年，Twitter や Facebook などのソーシャルメディアの投稿から，個人や社会の意見や感情を抽出したいというニーズが高まっている．スポーツの試合，総選挙，テレビ放送などのイベントや，商品，会社，俳優などのエンティティに言及する投稿を収集し，そのデータに対して感情分析や情報抽出を行えば，イベントやエンティティに関するマーケティングに活用できる．

このような分析において典型的に行われるのは，イベントやエンティティに言及する投稿を収集するタスクを情報検索と見なし，ソーシャルメディアの投稿をクエリで検索することである．例えば，ドラマ「あまちゃん」に関する Twitter 上での評判を調べたい場合は，「あまちゃん」や「#amachan」を検索クエリとしてツイート検索を行うのが常套手段である．しかし，この方法では「アキ可愛かった」という投稿のように，「あまちゃん」やハッシュタグを使わずに，ドラマの内容に言及するツイートを取得できない．

次善の策として，イベントの期間内（「あまちゃん」の例では放送時間内）にイベントに関するキーワードを投稿したユーザは，そのイベントに参加している（放送を見ている）と仮定して，分析対象をキーワードを含まない周辺のツイートに拡大することが考えられる．しかし，ソーシャルメディアは元々コミュニケーションツールであるため，イベントの期間内でも友人と別の話題の会話をしたり，イベントの参加を途中で打ち切って，別の話題を展開することがある．

そこで，本論文では特定のイベントに言及するツイートを自動的に判別する手法を提案する．提案手法の概要を図 1 に示す．本研究では，イベントに関するクエリを用いて，ソーシャルメディア上の投稿を収集する．収集された投稿に基づき，そのイベントに言及しうる関連キーワードを自動的に抽出し，ツイートのイベントへの関連度を計算する．イベントの中で起こる突発的な盛り上がり事象を捉えるため，時間的に近接する投稿の内容を考

*第 1 著者と第 2 著者は本論文において等しい貢献をした

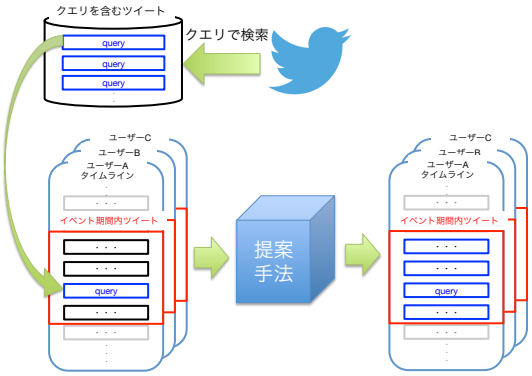


図 1: 提案手法の流れ

慮した関連度や，ユーザの投稿の一貫性を考慮した関連度を計算し，キーワードでは検索されなかったがイベントに言及しているツイートを認識する．テレビ番組を視聴するという行為をイベントへの参加と見なし，3 つのテレビ番組に関する正解データを作成し，提案手法の性能を評価した．実験結果から，提案手法によるキーワードの拡張，時間的に近接する投稿，ユーザの話題の一貫性のいずれも性能の向上に寄与することを実証した．

## 2 関連研究

小林ら [1] は，イベント開始直後のツイートに着目して各ユーザに対し「観覧」「不観覧」のラベル付けを行い，「観覧」ラベルの付与されたユーザの使用頻度が高い単語から作成した特徴ベクトルを SVM で識別することにより，ハッシュタグによらない検出を実現している．しかし，ユーザ単位でイベントの特徴を判断するため，観覧ユーザのツイート傾向に影響されやすく，ノイズが混ざりやすい．

山本ら [2] は，対象とするイベントをテレビ番組に限定し，字幕テキストを用いることで時間とともに推移する番組内容に則した特徴語を抽出している．しかし，字幕テキストのようにイベントの内容を時系列で記録したデータが必要であるため，常に利用できるわけではない．

Magdy ら [3] は各イベントに固定のクエリを結びつけるのではなく，時間毎に tf-idf 値を算出し直すことでイベント内のサブイベントの変化を追う手法を提案した．

単語の関連度を時間毎に求める点では本手法と類似しているが，ユーザ内の話題の一貫性は考慮されない．

## 3 提案手法

本節では提案手法を説明する．本手法は，対象のイベントに関するクエリ $q$ を用いて収集したツイートを出発点として，ツイートの拡張を進めていく．まず，収集したツイートの中に含まれるキーワードを分析し，対象のイベントとの関連度を計算する（3.1 節）．さらに，対象のイベントの中で突発的に盛り上がる事象に対応するため，各ツイートから時間的に近接するツイートに含まれるキーワードに基づき，ツイートとイベントの関連度を再計算する手法を提案する（3.2 節）．さらに，ユーザの投稿内容の一貫性をするため，各ツイートの前後のツイートを考慮した関連度を設計する（3.3 節）．

### 3.1 キーワードの自動拡張に基づく関連度

ある発言がイベントに関連するかどうかは，その発言がイベント名を含むかどうかで調べることができる．しかしながら，イベントは多くの要素から成り立っている．例えば開催地，主催者，出演者などのイベント固有の特徴や，発言や出来事などのイベント内の個別の事象への言及から，発言がイベントに関するものなのか判別できる．山本ら [2] の手法では，イベントに関する知識を活用して発言の識別を行っているが，このような知識を事前に獲得しておくことは難しい．

本研究では，Kaji ら [4] の手法を参考に，イベントに関して言及しうる表現の拡張を行った．まず，ある時間帯（例えばイベントの開催期間内）に発信されたツイートの中で，クエリ $q$ に言及しているツイートの集合を $D_+$ とする．すなわち，$D_+$ はある時間帯の範囲内でクエリ $q$ に合致するツイートを検索することで得られる．$D_+$ に含まれるツイートを発信したユーザの集合を $U_+$ とする．さらに，同じ時間帯のツイートの中で，クエリ $q$ に合致するツイートを 1 度も行わなかったユーザの集合を $U_-$ とし，ユーザ集合 $U_-$ からのツイートの集合を $D_-$ とする．すなわち，$D_-$ はイベント $q$ に言及していない（と推定される）ユーザからのツイートの集合である．

この 2 つのツイート集合を用い，ある単語 $w$ のイベントとの関連度を式 1 で定義する．

$$\mathrm{sw}(w) = \mathrm{PMI}(w, D_+) - \mathrm{PMI}(w, D_-) \tag{1}$$

ただし，$\mathrm{PMI}(w, D)$ は相互情報量で，

$$\mathrm{PMI}(w, D) = \log \frac{P(w, D)}{P(w)P(D)} \tag{2}$$

であるから，式 1 は式 4 で表される．

$$\begin{aligned} \mathrm{sw}(w) &= \log \frac{P(w|D_+)}{P(w|D_-)} \\ &\approx \log \frac{\frac{\#(w,D_+)+\alpha}{|D_+|}}{\frac{\#(w,D_-)+\beta}{|D_-|}} \end{aligned} \tag{3, 4}$$

ここで，$\#(w, D_+)$ は $D_+$ のツイート集合の中で単語 $w$ を含むツイートの数，$\#(w, D_-)$ は $D_-$ のツイート集合の中で単語 $w$ を含むツイートの数，$|D_+|$ と $|D_-|$ はそれぞれ，$D_+$ と $D_-$ に含まれるツイートの数である．また，$\alpha$ と $\beta$ は $\#(w, D_+)$ や $\#(w, D_-)$ が 0 になる場合に対応するもので，次式で定義する．

$$\alpha = \frac{|D_+|}{|D_+| + |D_-|}, \tag{5}$$

$$\beta = \frac{|D_-|}{|D_+| + |D_-|} \tag{6}$$

式 4 で求められる関連度は，大きい値を持つ単語ほどイベントへの関連が深い．関連度 0 はイベントとの無相関，負の値はイベントとの逆相関を表す．

式 4 は，イベントに関連しそうな単語を自動的に拡張するものである．これに基づき，ツイート $d$ のイベントとの関連度 sd(d) を，そのツイートが含む単語 $w \in d$ の関連度スコア $\mathrm{sw}(w)$ の平均として算出する．

$$\mathrm{sd}(d) = \frac{1}{|d|} \sum_{w \in d} \mathrm{sw}(w) \tag{7}$$

### 3.2 時間的近接性に基づく関連度の再計算

テレビ番組で出演者が面白い発言をすると，その発言内容が Twitter 上で盛り上がることがある．このようなイベント中の盛り上がりは突発的な事象であるため，式 4 のようなイベント全体を通したスコア付けでは捉えにくい．そこで，あるツイート $d$ のイベントとの関連度を計算する際，時間的に近接するツイートの内容に基づきツイートの関連度を再計算する．

あるツイート $d$ が発信された時刻を $t$ とし，そのツイートの前後 $\Delta$ 秒の期間 $[t - \Delta, t + \Delta]$ を $p$ と書く．あるユーザ $u$ が期間 $p$ の間につぶやいたツイート集合を $D_{u,p}$ と書き，その期間におけるユーザ $u$ とイベントの関連度を次式で計算する．

$$\mathrm{su}(u, p) = \frac{1}{|D_{u,p}|} \sum_{d \in D_{u,p}} \mathrm{sd}(d) \tag{8}$$

式 8 の関連度は正規化されていないため，次式で正規化する．

$$\mathrm{su}'(u, p) = \begin{cases} max\left(0, \frac{\mathrm{su}(u,p)}{\max_{u' \in U_+} \mathrm{su}(u',p)}\right) & (u \in U_+) \\ max\left(0, \frac{\mathrm{su}(u,p)}{\min_{u' \in U_-} \mathrm{su}(u',p)}\right) & (u \in U_-) \end{cases} \tag{9}$$

式 9 により，$u \in U_+$ であるユーザ $u$ はイベントへの関連度が $[0,1]$ で表され，$u \in U_-$ であるユーザ $u$ はイベントへの非関連度が $[0,1]$ で表される．

式 4 を修正し，キーワード $w$ のイベントへの関連度を，そのキーワードをつぶやいたユーザの関連度から再計算する．

$$\mathrm{sw}_t(w) = \log \frac{\frac{\sum_{u\in U_+} |w\in D_{u,p}|\cdot \mathrm{su}'(u)+\alpha'}{\sum_{u\in U_+}|D_{u,p}|}}{\frac{\sum_{u\in U_-} |w\in D_{u,p}|\cdot \mathrm{su}'(u)+\beta'}{\sum_{u\in U_-}|D_{u,p}|}}, \quad (10)$$

$$\alpha' = \frac{\sum_{u\in U_+}|D_{u,p}|}{\sum_{u\in (U_+\cup U_-)}|D_{u,p}|}, \quad (11)$$

$$\beta' = \frac{\sum_{u\in U_-}|D_{u,p}|}{\sum_{u\in (U_+\cup U_-)}|D_{u,p}|} \quad (12)$$

ただし，$|w \in D_{u,p}|$ はユーザ $u$ の期間 $p$ のツイートの中でキーワード $w$ を含むツイートの数を表す．

最後に，再計算したキーワードのスコアを用いて，ツイートの関連度スコアを再計算する．

$$\mathrm{sd}_t(d) = \frac{1}{|d|}\sum_{w\in d}\mathrm{sw}_t(w) \quad (13)$$

これまでの議論は，関連度スコアの再計算を行いたいツイート $d$ に依存して期間 $p$ やキーワードスコアの再計算を行っていた．このような再計算を，全てのツイートに関して個別に実施し，時間的近接性に基づくツイートの関連度の再計算を行う．

### 3.3 前後ツイートを考慮した関連度

あるユーザが短時間の間につぶやく話題は同じであることが多いが，式 13 が考慮するのはユーザを横断した話題の類似性であり，ユーザ内での話題の一貫性は考慮されていない．そこで，ユーザのツイートの中で，ツイート $d$ の直前のツイート $d_{-1}$ と直後のツイート $d_{+1}$ の関連度を考慮したスコア付け手法を提案する [1]．最終的なツイートの関連度スコアは，3.1 節と 3.2 節の関連度の式を統合し，式 14 で計算する．

$$\begin{aligned} & \mathrm{score}(d) \\ = \ & \mathrm{sd}(d) + \mathrm{sd}_t(d) \\ + \ & \max\left(0, 1-\log_\theta \delta(d,d_{-1})\right)\left\{\mathrm{sd}(d_{-1})+\mathrm{sd}_t(d_{-1})\right\} \\ + \ & \max\left(0, 1-\log_\theta \delta(d_{+1},d)\right)\left\{\mathrm{sd}(d_{+1})+\mathrm{sd}_t(d_{+1})\right\} \end{aligned} \quad (14)$$

ここで，$\delta(d, d_{-1})$ はツイート $d$ と $d_{-1}$ の時間差（秒）で，$\theta$ は時間的に遠いツイートの影響力を打ち切るための閾値である．式 14 の $\max(0, 1-\log_\theta \delta(d,d_{-1}))$ は，2 つのツイート $d, d_{-1}$ の時間差が短いほど大きな値をとり，時間差が $\theta$ 以上になると 0（影響力なし）になる．

[1] 厳密には，$d_{-1}$ はツイート $d$ よりも時間的に古く，リプライではないツイートの中で最もツイート $d$ に時間的に近いもの，$d_{+1}$ はツイート $d$ よりも時間的に新しく，リプライではないツイートの中で最もツイート $d$ に時間的に近いものを採用した．

## 4 評価実験

### 4.1 実験データ

本研究では多数のユーザが同時に投稿を行うイベントとして，3 つのテレビ番組を対象に実験を行う．実験対象のテレビ番組は，視聴率が高く Twitter への投稿数も多いと思われる「THE MANZAI」「あまちゃん」「携帯大喜利」とする．各番組の放送時間内に 5 回以上つぶやいたユーザのツイートを実験データとした．実験データの中で，番組のクエリに合致するツイートを 1 回以上つぶやいたユーザをランダムに抽出し，そのユーザのツイートから評価データを作成した．評価データに含まれるツイートを人間が読み，番組に関するツイートかどうか，ラベル付け作業を行った．表 1 に，実験データ，評価データの概要を示した．

### 4.2 実験設定

今回の実験では，$\mathrm{sd}_t(d)$ の計算における時間的近接の範囲 $\Delta = 300$ 秒，前後のツイートで影響力が及ぶ範囲 $\theta = 300$ 秒 とした．提案した手法の貢献を分析するため，以下の手法でイベントとの関連性を認識した．

- ツイートをクエリのみで検索したもの
- キーワードの自動拡張に基づく関連度のみを考慮した手法（$sd(d)$ でスコア付け）
- 時間的近接性に基づく関連度も併せて考慮した手法（$sd(d) + sd_t(d)$ でスコア付け）
- ユーザの前後のツイートも考慮した手法（$score(d)$）

### 4.3 実験結果

表 2 に実験結果を示す．どの番組に対しても，クエリによる検索を基準として，キーワードの自動拡張，時間的近接性，前後のツイートを考慮することにより性能の改善が見られた．特に，キーワードの自動拡張と前後のツイートによる性能の改善が著しい．

どの番組においても，全ての要素を考慮した提案手法は，適合率を約 90%に保ちつつ，クエリによる検索のみのベースライン手法と比べて，1.64 倍〜3.2 倍のツイートを収集することができた．

本手法によって「あまちゃん」に関するツイートとして獲得できるようになったものの例を以下に示す．

- おお、甲斐さん
- 焼うどんwww
- 春子おおおおおおおおおおおおおおおおおおおお
- だめだ、泣かずには見られん ( ´ Д⊂ヽ
- ここ最高やで。。

表 1: 実験データの詳細

| | | THE MANZAI（バラエティ） | 着信御礼！ケータイ大喜利（バラエティ） | あまちゃん（ドラマ） |
|---|---|---|---|---|
| 番組放送日時 | | 2014/12/14 17:30 〜 2014/12/14 19:58 | 2014/9/21 00:05 〜 2014/9/21 00:50 | 2013/9/28 8:00 〜 2013/9/28 8:15 |
| 収集した | ユーザ数 | 236,158 | 99,097 | 10,884 |
| | ツイート総数 | 3,142,149 | 1,203,015 | 103,568 |
| 評価データに用いた | ユーザ数 | 130 | 115 | 101 |
| | ツイート総数 | 2,298 | 1,867 | 1,082 |
| 検索に用いたクエリ | | THE MANZAI<br>ザ・マンザイ<br>#manzai<br>#fujiTV | ケータイ大喜利<br>#ケータイ大喜利<br>#nhk | あまちゃん<br>#あまちゃん<br>#nhk |

表 2: 実験結果

| | THE MANZAI | | | 着信御礼！ケータイ大喜利 | | | あまちゃん | | |
|---|---|---|---|---|---|---|---|---|---|
| | 適合率 | 再現率 | F1 | 適合率 | 再現率 | F1 | 適合率 | 再現率 | F1 |
| クエリのみ | 1.0000 | 0.2792 | 0.4365 | 0.9282 | 0.1934 | 0.3201 | 0.9778 | 0.5858 | 0.7327 |
| ＋ キーワード拡張 | 0.8966 | 0.8662 | 0.8811 | 0.8800 | 0.5641 | 0.6875 | 0.9445 | 0.8817 | 0.9120 |
| ＋ 時間的近接性 | 0.9006 | 0.8644 | 0.8821 | 0.8781 | 0.5694 | 0.6909 | 0.9391 | 0.8905 | 0.9142 |
| ＋ 前後ツイート | 0.8943 | 0.9135 | 0.9038 | 0.8970 | 0.6325 | 0.7419 | 0.9246 | 0.9615 | 0.9427 |

この例が示すように，番組の登場人物や状況に関するキーワードを自動的に獲得できている．また，視聴者が共通に抱く感想も提案手法で捉えられていることが分かる．

一方，本手法では獲得できなかったツイートの例を以下に示す．

- うそだろ
- うおおおおお
- これはキツイな…
- 俺だったら無理だわw
- (( ^ ω ^ ≡ ^ ω＜
- なにそれwwwww

この例が示すように，本手法では獲得できないツイートの特徴は以下の通りである．

- 顔文字や奇声など，形態素解析が困難である表現が含まれている
- 個人の意見・感想が述べられている
- ツイートの内容が薄い

テレビ番組「着信御礼！ケータイ大喜利」の F1 値が低くなっているは以上の要因によるものである．

## 5 おわりに

本論文では，特定イベントに対する評判分析を行うために，イベントに関連するキーワードの自動的な拡張を行い，イベント内容の時間的変化と前後ツイートを考慮する手法を提案した．実験の結果，単純な検索では収集できなかったツイートも獲得できることが示された．今後の課題として，テレビ番組だけでなく他の種類のイベントに関しても評価を行うことが挙げられる．

**謝辞** 本研究は，文部科学省科研費課題 23240018 の一環として行われた．また JST 戦略的創造研究推進事業「さきがけ」から部分的な支援を受けて行われた．

## 参考文献

[1] 小林尊志, 野田雅文, 出口大輔, 高橋友和, 井手一郎, 村瀬洋. Twitter における実況書き込み検出手法の検討. 映像情報メディア学会技術報告, Vol. 34, No. 25, pp. 129–130, 2010.

[2] 山本祐輔, 浅井洋樹, 上田高徳, 秋岡明香, 山名早人. テレビ番組に対する意見をもつ twitter ユーザのリアルタイム検出. 第 5 回データ工学と情報マネジメントに関するフォーラム, pp. C1–4, 2013.

[3] Walid Magdy and Tamer Elsayed. Adaptive method for following dynamic topics on twitter. In *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*, pp. 335–345, 2013.

[4] Nobuhiro Kaji and Masaru Kitsuregawa. Building lexicon for sentiment analysis from massive collection of HTML documents. In *Proceedings of EMNLP-CoNLL 2007*, pp. 1075–1083, 2007.