

# Twitter 上の位置情報付き画像を利用したリアルタイムイベント画像検出

金子 昂夢\*1      柳井 啓司\*2  
Takamu KANEKO      Keiji YANAI

電気通信大学 大学院情報理工学研究所 総合情報学専攻  
Department of Informatics, The University of Electro-Communications

近年、スマートフォンのような位置情報を扱えるデバイスの普及に伴い、撮影した画像に位置情報を付加することが容易となった。また、Twitter の普及により、自分が今何をしているのか、どこにいるのかといったことを投稿する人が増加してきた。Twitter では本文の他に画像や位置情報を添付でき、撮影した位置情報付き画像を即座に投稿することができる。本研究では、これらの位置情報付き画像を持った投稿からリアルタイムイベント画像検出を試みる。まず、対象をより小さな地域に分割し、各地域における投稿をリアルタイムに監視し、前日の状態と比較してキーワードを検出した。そして、キーワードを含む投稿の各画像から画像特徴量を用いてクラスタリングを行い、その結果からイベントの代表的な画像を選出し、地図上に表示させた。実験では日本で投稿されたツイートをを用いて実験を行った結果、リアルタイムに投稿された内容から関西地方で雪が観測されたことが検出され、画像からその様子を視覚的に捉えることができた。

## 1. はじめに

近年、スマートフォンのような位置情報を扱えるデバイスの普及に伴い、撮影した画像に位置情報を付加することが容易となった。また、ツイートと呼ばれる短文を投稿できるマイクロブログサービス Twitter の普及により、自分が今何をしているのか、どこにいるのかといったことを投稿する人が増加してきた。ツイートには本文の他に画像や位置情報を添付でき、撮影した位置情報付き画像を即座に投稿することができる。このため Twitter には多くの位置情報付き画像がテキストデータと共に投稿され、これらのデータはリアルタイム性という重要な性質を持つ。Twitter に投稿された位置情報付き画像ツイートの例を図 1 に示す。



図 1: 位置情報付き画像ツイートの例

本研究では、Twitter に投稿された位置情報付き画像ツイートからリアルタイムなイベント画像の検出を試みる。イベントを検出し、その情報を画像と共に地図上に表示することで、今どこで何が起きているのかを視覚的に捉えることを目標とする。

## 2. 関連研究

Twitter はテキストマイニング対象として広く研究されており、イベントの検出を試みた研究も多くある。

榊ら [1] は、Twitter のユーザを実世界のイベントを監視するソーシャルセンサと捉えたモデルを構築し、イベントの検出とその位置の推定を行うシステムを作成した。

Lee ら [2] はより局所的なイベントを検出するため、対象地域をいくつかのより小さな地域に分割し、各地域の通常の状態を推測する。そして、通常の状態とは異なるツイートが多く投稿されたとき、イベントが発生したとみなしている。

中地ら [3] はあらかじめ特定のキーワードや期間を設けて Twitter から位置情報付き画像ツイートを収集し、画像を解析することで異なる時間や位置において投稿される画像の違いを示した。

さらに、位置情報付き画像ツイートを利用した研究として、柳井らの World Seer [4] がある。Twitter に投稿された位置情報付き画像をリアルタイムに監視し、地図上にマッピングすることでどのような写真がどこで撮られたのかを瞬時に視覚化され閲覧することができるシステムであった。

## 3. 手法の概要

本研究では、まずバッチ処理するシステムを構築し、その後システムをオンライン化することでリアルタイム検出を行う。検出には任意のイベントキーワードを自動的に抽出し、キーワードを用いて収集した画像を解析することで、画像により視覚的に捉えることができるようにする。本研究の手法の流れを以下に示す。

1. 特定の地域で頻出するキーワードの抽出
2. 抽出されたキーワードの統合・補完
3. キーワードに対する位置情報付き画像ツイートの収集
4. 画像から Bag-of-Features とカラーヒストグラムを生成
5. 2 つの画像特徴量を用いてクラスタリング
6. クラスタリング結果から代表画像選出・中心座標計算
7. 検出されたイベントを地図上に表示

まず、対象領域をより小さな地域に分割し、各地域で前日の状態と比較して得られるスコアが一定以上となるキーワード

連絡先: kaneko-t@mm.inf.uec.ac.jp\*1

連絡先: yanai@cs.uec.ac.jp\*2

を抽出する。抽出されたキーワードは、同日・同地域で検出された同じイベントに関するものを統合し、キーワードを含むツイートから前後の文字の補完を行う。

次に、検出されたイベントのキーワードを用いて位置情報付き画像ツイートを収集し、各画像から SURF による Bag-of-Features と RGB カラーヒストグラムを生成する。

そして、画像を Ward 法によりクラスタリングし、その結果から得られるスコアによりクラスタをランキングする。最もスコアが高いクラスタからイベントの代表的な画像を選出し、スコアが一定以上のクラスタに含まれる位置情報付き画像ツイートからイベントの中心座標を計算して地図上に表示する。またスコアが一定未満のクラスタはノイズ画像群とし、イベントから除外する。

## 4. 手法の詳細

リアルタイム検出における各ステップの処理について述べる。

### 4.1 キーワードの検出

視覚的なイベントの検出を行うために、位置情報付き画像ツイートの収集に用いるイベントのキーワードを検出した。

#### 4.1.1 キーワードの抽出

イベントのキーワードを検出するために、まずツイートの本文に対して MeCab を用いて形態素解析を行った。そして、抽出された各ツイートの名詞のうち、特定の地域で前日の状態と比較して得られるスコアが一定以上となるものを 1 日ごとに探索し、イベントのキーワードとして抽出した。

対象地域の様々な地域から抽出されるように対象を緯度・経度それぞれ 1 度ずつのグリッドにより分割し、各地域に重みを付けた。分割された各地域について、データセットの全ての位置情報付き画像ツイートから、そのグリッド内でツイートしたユーザ数をカウントし、最も出現したユーザ数が多い地域が 1 となるように以下の式 1 により重みを付けた。

$$W_{i,j} = \frac{M + s}{N_{i,j} + s} \quad (1)$$

式 1 において、 $i, j$  はそれぞれ緯度・経度によるグリッドを表し、 $M$  は最も出現するユーザ数が多い地域のユーザ数、 $N_{i,j}$  はグリッド  $i, j$  に出現するユーザ数、 $s$  は標準偏差である。この重みによりスコアを補正することでユーザが少ない地域からでもキーワードが抽出されるようにした。

そして、タグに登録された各単語について、それぞれの地域でその単語をタグに含むツイートを投稿したユーザ数と前日のそれとの差に注目している地域の重みを乗じた値をスコアとし、スコアが一定以上となるものを抽出した。

$$S_{k,d,i,j} = (N_{k,d,i,j} - N_{k,d-1,i,j})W_{i,j} \quad (2)$$

式 2 において、 $k$  は各単語、 $d$  は注目している日、 $N$  はそれぞれの条件で単語  $k$  をタグに含むツイートを投稿したユーザ数である。今回の実験では、閾値を 50 としてイベントのキーワードを抽出し、キーワードによる位置情報付き画像ツイートの収集は、キーワードが抽出された日と地域に注目して行った。

#### 4.1.2 キーワードの統合・補完

形態素解析では単語を最小単位まで分割するため、同日・同地域で検出された各キーワードについて、同じイベントに関するキーワードの統合を行った。一方のキーワードをタグに含むツイートの、もう一方のキーワードが 50% 以上含まれていれば、同じイベントに関するキーワードとして統合した。統合

されたキーワード群に対し新たなキーワードを統合する場合、いずれかの組で条件を満たせば同じイベントに関するキーワードとして統合した。統合されたキーワードのうち、そのイベントのキーワードは最も多くのツイートに含まれるものとした。

また、形態素解析に用いる辞書により単語は切り取られるため、抽出されたキーワードを含むツイートを利用して、キーワードの補完を行った。元のキーワードの一つ前の文字が 80% 以上同じ文字ならば、この文字でキーワードを補完し、さらに補完されたキーワードの一つ前の文字を調べ、再帰的に補完を行った。キーワードの後ろの文字についても同様に補完を行った。

### 4.2 画像のクラスタリング

キーワードを用いて収集した画像から類似する画像群を抽出するため、各画像から SURF による Bag-of-Features と RGB カラーヒストグラムを生成し、クラスタリングを行った。クラスタリングには Ward 法を用い、閾値を設定することで任意のクラスタ数に画像が分類されるようにした。

Ward 法はクラスタ内の各要素からその中心までの距離を最小化する手法であり、凝集型階層的クラスタリングの一種である。全ての要素が独立したクラスタ状態から始め、以下の式 3 により定義されるエラーの増加量が最小となるクラスタの組を逐次的に併合する。

$$d(C_1, C_2) = E(C_1 \cup C_2) - E(C_1) - E(C_2) \quad (3)$$

ただし、通常  $E(C)$  はクラスタ  $C$  の全ての要素から中心までの距離の二乗の総和であるが、今回使用する画像特徴量は二つあるので、以下の式 4 のように  $E(C)$  を定義した。

$$E(C) = \sum_{x \in C} ((x_{BoF} - \bar{x}_{BoF})^2 w_{BoF} + (x_{Color} - \bar{x}_{Color})^2 w_{Color}) \quad (4)$$

式 4 において、 $x_{BoF}$  は画像  $x$  の Bag-of-Features、 $x_{Color}$  は画像  $x$  のカラーヒストグラム、 $\bar{x}$  はクラスタ  $C$  における各特徴量の中心、 $w$  は各特徴量の重みでそれぞれの次元数である。

本研究では任意のクラスタ数に分類されるように、エラーの増加量  $d(C_1, C_2)$  に対して閾値を設定し、 $d(C_1, C_2)$  が閾値よりも大きくなるか、クラスタ数が 1 となることをクラスタリングの終了条件とした。今回の実験では閾値を 300 とした。

### 4.3 視覚的な検出

クラスタリング結果により、視覚的に統一性があり画像枚数が多いクラスタに限定することでイベントを視覚的に表すことができる画像群を抽出し、地図上にイベントの検出結果を表示した。

各クラスタについて、より画像の類似度が高く、分類された画像枚数が多いクラスタが高いスコアとなるように、以下の式 5 によりスコアを付けた。

$$V_C = \frac{n_C^2}{E(C)} W_{i,j} \quad (5)$$

式 5 において、 $n_C$  はクラスタ  $C$  に分類された画像枚数、 $W_{i,j}$  は式 1 によって得られた、キーワードが検出された地域の重みである。この値が大きいほどそのクラスタのイベントに対する視覚的関連性が高く、小さいほどノイズである可能性が高いと言える。

このスコアが閾値以上のクラスタをイベントを視覚的に表すことができる画像群とし、閾値未満のクラスタはノイズ画像群

としてイベントから除去した。さらに、各イベントにおいて、スコアが最も高いクラスタの中でクラスタ中心に最も近い画像をイベントの代表的な画像として選出し、閾値以上のクラスタに含まれる位置情報付き画像ツイートからイベントの中心座標を求めた。そして、得られた中心座標を用いて地図上にマーカーを設置し、イベントの情報と代表画像をマーカーに対応付け表示させた。

## 5. 実験

まず、データの集合を与えバッチ処理を行うことでどのようなイベントと代表画像が検出されるかについて実験を行った。次に、実際に Twitter のタイムラインをリアルタイムに監視し、各ステップの処理をオンラインで行った場合について実験を行った。

### 5.1 バッチ処理実験

#### 5.1.1 データセット

バッチ処理実験を行うにあたって、Twitter に投稿された位置情報付き画像ツイートからデータセットを作成した。日本国内で 2011 年 2 月 10 日から 2012 年 9 月 30 日までに投稿されたものであり、約 3,000,000 件の位置情報付き画像ツイートである。

#### 5.1.2 キーワードの検出結果

テキスト解析により抽出され、統合・補完が行われたキーワードの一部を表 1 に示す。表中の「地域」は左からグリッドにより分割された各地域の南端、北端、西端、東端の緯度・経度を表し、「重み」は各地域の重みである。

#### 5.1.3 クラスタリング

検出されたキーワードを用いて収集した画像を画像特徴量を用いて、Ward 法によりクラスタリングを行った。

クラスタリング結果の例それぞれ図 2、図 3 に示す。図中の各クラスタの上部に記載されている“num”はそのクラスタに分類された画像枚数であり、“b\_score”と“c\_score”はそれぞれ Bag-of-Features とカラーヒストグラムの各画像からクラスタ中心までの距離の二乗に特徴量の次元を乗じた値の総和で、この値が小さいほどその特徴量についての類似性の密度が高いことを表している。また、“weight”はキーワードが検出された地域の重みを表し、“score”がこれらの値から計算されたクラスタのスコアである。

クラスタリング結果により、スコアが閾値未満のクラスタをノイズ画像群としてイベントから除去された。

図 2 の「花火」の例は、日本データセットで 2011 年 8 月 6 日に東京で検出されたイベントであり、最も高いスコアを持つ最上部のクラスタを見ると花火が写った画像が多く含まれている。

図 3 の「桜」の例は、スコアが閾値以上である上 2 つのクラスタを見ると、両方とも桜が写り、このイベントに関する画像であるが、背景の色の違いにより日中に撮影されたものと夜に撮影されたものに分類され、時間によるイベントの様子の違いが示された。

#### 5.1.4 検出結果の表示

検出されたイベントを地図上に表示するため、スコアが最も高いクラスタの中でクラスタ中心に最も近い画像を代表的な画像として選出し、スコアが閾値以上のクラスタに含まれる位置情報付き画像ツイートから中心座標を求めた。

検出されたイベントについて、地図上にマーカーを設置し、いくつかのマーカーをクリックした例を図 4 に示す。Web 上

のシステムでは、表示された代表画像をクリックすることで、そのイベントのクラスタリング結果が表示され、イベントに関するその他の画像を閲覧することができる。

最終的なイベントの検出結果は検出されたイベント数が 258、代表画像の適合率は 65.5%となった。

### 5.2 リアルタイム処理実験

本手法を用いて、Twitter からのリアルタイム検出実験を行った。Twitter Streaming API を利用してリアルタイムにツイートを取得し、一時的にデータをプールしながら、100 秒ごとに検出スコアの閾値を超えるキーワードを調べた。ツイートの取得と検出を並列に行い、さらにキーワードが検出された場合、新たに画像処理を行うプロセスを生成することで大量の投稿に対してもリアルタイムに処理できるようにした。実験を行うにあたり、キーワードの検出スコアの閾値を 30 とし、キーワードが検出された段階で画像の解析を行い、代表画像を抽出した。2014 年 3 月 10 日を対象に実験を行った結果、この日、関西地方を中心に雪が観測されたため、「雪」というキーワードが多く検出された。今回の実験で検出されたイベントを表 2 に示す。また、図 5 は検出された「雪」の例であり、地図上の画像が選択された代表画像である。

## 6. まとめ

本研究では、Twitter に投稿された位置情報付き画像ツイートからリアルタイムなイベント画像検出を行った。Twitter のタイムラインをリアルタイムに監視し、前日と比べ出現頻度が一定以上上昇したキーワードを検出し、キーワードをテキストを含む画像を解析することでイベントの代表的な画像を抽出した。リアルタイム処理実験の結果、バッチ処理システムをオンライン化することで、関西地方での雪の降雪を検出することができた。

今後の課題として、まず、検出するグリッドの大きさと時間の刻みを可変にし、より柔軟にイベントの検出が行えるようにする。また、代表的な画像の選出方法を改良し、その精度を上げると共に、同キーワードのイベントについて、時間や地域における画像の違いを抽出することが考えられる。

## 参考文献

- [1] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proc. of the International World Wide Web Conference*, pp. 851–860, 2010.
- [2] R. Lee and K. Sumiya. Measuring geographical regularities of crowd behaviors for twitter-based geo-social event detection. In *Proc. of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks*, pp. 1–10, 2010.
- [3] Y. Nakaji and K. Yanai. Visualization of real world events with geotagged tweet photos. In *Proc. of IEEE ICME Workshop on Social Media Computing (SMC)*, 2012.
- [4] K. Yanai. World seer: a realtime geo-tweet photo mapping system. In *Proc. of ACM International Conference on Multimedia Retrieval*, 2012.

表 1: 検出されたイベントの一部

キーワード	日付	地域 (lat,lng)	重み	スコア
雪	2011/2/11	34 ~ 35,135 ~ 136	1.96	135.5
地震	2011/3/11	35 ~ 36,139 ~ 140	1	55
花火大会	2011/8/6	34 ~ 35,135 ~ 136	1.96	149.2
台風	2011/9/21	35 ~ 36,139 ~ 140	1	62
月食	2011/12/10	34 ~ 35,135 ~ 136	1.96	84.4
大晦日	2011/12/31	35 ~ 36,139 ~ 140	1	68
初日の出	2012/1/1	35 ~ 36,139 ~ 140	1	84
マラソン	2012/2/26	35 ~ 36,139 ~ 140	1	77
桜	2012/4/28	37 ~ 38,140 ~ 141	4.18	121.4
スーパーmoon	2012/5/5	35 ~ 36,139 ~ 140	1	93
金環日食	2012/5/21	35 ~ 36,139 ~ 140	1	314
祇園祭	2012/7/14	35 ~ 36,135 ~ 136	3.46	104
スカイツリー	2012/7/28	35 ~ 36,139 ~ 140	1	96
精霊流し	2012/8/15	32 ~ 33,129 ~ 130	4.69	51.6
送り火	2012/8/16	35 ~ 36,135 ~ 136	3.46	104



図 3: 「桜」のクラスタリング結果

表 2: 3月10日に検出されたイベント

キーワード	位置 (lat,lng)	時刻 (h:m:s)
雪	43.0,141.4	8:41:23
雪	35.0,136.1	8:51:25
富士山	35.3,138.9	12:9:41
雪	36.6,136.6	12:11:57
雪	35.0,135.7	12:30:20
雪	35.7,137.9	12:35:19
雪	34.5,136.7	12:38:40
雪	34.9,135.8	12:48:45



図 4: 検出結果の表示例

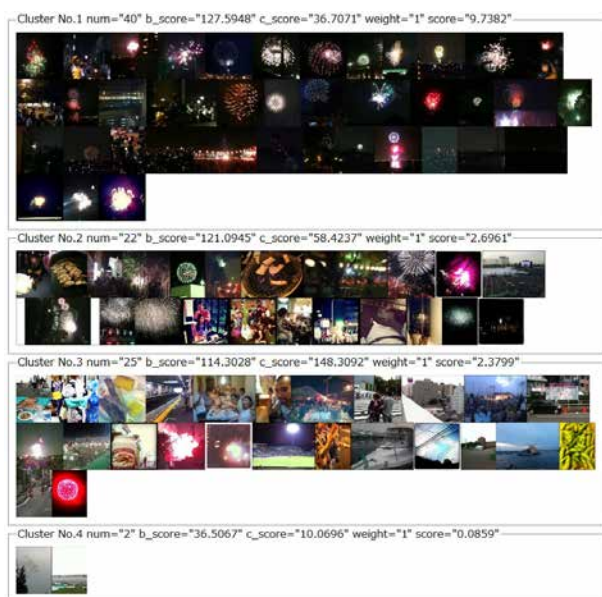


図 2: 「花火」のクラスタリング結果



図 5: リアルタイム処理により検出された「雪」の例