Twitter メッシュデータ収集・視覚化システム

藤田秀之

Twitter mesh data acquisition and visualization system

Hideyuki Fujita

Abstract: Mobile social media is generating good data for analyzing human behaviors and context of areas. In this research, we developed a distributed system for collecting tweet data aggregated to standard mesh system. The proposed method can collect several times as much data as common methods. We also developed a spatio-temporal visualization tool for them.

Keywords: ソーシャルメディア（social media），ツィッター（Twitter），視覚化（visualization），アニメーション（animation），データ共有（data sharing）

1. 序論

モバイル・ソーシャルメディアのデータは，人々の行動や地域の状態を分析するための新たな情報源として期待されている．代表例である Twitter は，ユーザが Tweet(つぶやき)と呼ばれる短文を投稿するサービスであり，マイクロブログとも呼ばれる．2011 年 7 月現在，Twitter 社発表のサービス利用状況は次のとおりである．

・1 日あたりの Tweet 数:2 億（うち 25%が日本人）
・アカウント数：2 億
・1 日あたりのアカウント増加数：60 万

短文のため更新頻度が高く，スマートフォンの普及に伴い，モバイル端末からの投稿も多い．位置情報付きで投稿することも可能である．

冒頭の応用に向け，大量のデータの収集・蓄積が必要だが，後述のとおり，従来の収集手法で取得可能な件数は十分とは言えず，大量のデータを継続的に収集するコストも小さくない．そこで本研究では，将来的に研究者に対するデータ提供サービスに発展させることを念頭に，Twitter のデータを地理的位置に関連付け，できるだけ多く収集するシステムの構築を目的とする．具体的には，Tweet データを標準地域メッシュ単位に集約し，分散環境で収集するシステムを構築する．また，本システムによるデータ収集実験を行い，主に時空間的な観点からデータの概要を示す．

藤田秀之 〒277-8568 千葉県柏市柏の葉 5-1-5
東京大学空間情報科学研究センター
Phone: 04-7136-4291
E-mail: fujita@csis.u-tokyo.ac.jp

Twitter のデータを利用した研究事例は数多く発表されているが，ソーシャルグラフと呼ばれる，ユーザ同士の関係性のネットワークに着目した，コミュニケーションやコミュニティに関する研究が多数を占める．地理的空間との関連性を主題とした研究はまだ少ないが，次のような事例がある．Sakaki et al. (2010)は，特定のイベント(地震と台風)に関する Tweet の時空間変化に着目し，リアルタイムのイベント検出や発生位置の予測を行う手法を提案している(表 1(1a)(1b))．Fujisaka et al. (2010)は，空間クラスタ内の Tweet 数の増減に着目したイベント抽出の手法を提案している(表 1(2))．van Liere al. (2010)は，他のユーザの Tweet を引用した Tweet(ReTweet)における両者の地理的距離に着目し，情報の拡散パターンを論じている(表 1(3))．

関連研究で利用された Tweet 数を表 1 に示す．(1)は特定のキーワードを含むデータのみのため，単純に比較できないが，空間的に十分な密度の Tweet データを得られないことは関連研究に共通の課題と考えられる．本研究は，現時点では対象とした空間範囲が狭いものの，関連研究と比較し，空間的に大幅に高密度にデータを取得している．

表-. 1. 関連研究と収集データ数

	範囲	期間	データ数	ユーザ数
(1a)	本州全域	60 日	地震関連 621	不明
(1b)	本州全域	8 日	台風関連 2,037	不明
(2)	日本全域	7 日	129,403	4,131
(3)	全世界	12 時間	13,339	6,424
本研究	都心部 20km 四方	14 日	3,476,059	216,430

2. データ収集手法

2.1 Twitter API

Twitter 社より多くの種類の Web API が提供されており，Tweet 本文とともに以下のような属性を取得可能である．

・つぶやき本文
・個々の Tweet に対する ID
・ユーザ ID
・宛先ユーザ ID（“@宛先ユーザ名”を指定して投稿した Tweet の場合）
・投稿日時
・プロフィール（現在地(自由記述)を含む(後述)）
・緯度経度（GeotaggingAPI を利用し投稿された場合のみ(後述)）

不特定多数のユーザのデータを取得する場合に利用可能な API は次の 2 種類である．概要をまとめる．

Streaming API

・サーバとの接続を維持している間，リアルタイムのサンプリングデータ(最大で全体の 10%程度)が送信されてくる
・空間フィルタを設定すると，緯度経度で指定した範囲の Tweet のみ取得可能
・空間フィルタ設定時に取得対象となる Tweet は，投稿時に専用の API(GeotaggingAPI)で位置情報を付与されたもののみである．これには，公式の Twitter クライアントから,位置情報通知機能を有効にして投稿されたもののみが該当する．（全 Tweet の 1%未満と言われている）

Search API

・過去約 5 日前までの Tweet を検索
・空間検索・時間検索が可能
・空間検索の対象となる Tweet は，前述の GeotaggingAPI で投稿されたデータに加え，Twitter 社が他の複数の方法で投稿場所を推定したデータである．

これらを利用して場所ごとにデータを取得する一般的な方法として,以下の 3 種類が挙げられる．利点・欠点をまとめる．

方法(a) Streaming API を利用．空間フィルタを設定し，リアルタイムデータをキャッシュ

利点
・各 Tweet の緯度経度情報も取得可能

欠点
・前述した条件により，件数が少ない

方法(b) Streaming API を利用．空間フィルタを利用せずリアルタイムデータをキャッシュし，取得した Tweet のユーザプロフィール情報の現在地名をジオコーディング

利点
・方法(a)と比較し件数が多い

欠点
・プロフィールの現在地名は自由記述のため，半数以上が「東京と千葉を往復」「週末は横浜」のような記述や，存在しない場所名等であり，ジオコーディングが困難か不可能である．

方法(c) Search API を利用．定期的に時空間検索して蓄積．

利点
・GeotaggingAPI 以外の位置情報も利用するため，空間検索の対象件数が多い（後述のとおり実際に取得できる件数は少ない）

欠点
・同一の検索条件に対し，1,500 件までしか取得できない
・空間検索の最小半径 1km，時間検索の最短期間は 1 日である，これらを同時に指定しても，都心部の多くの場所で結果が 1,500 件を大きく超えるが，前述の条件により，そのうち 1,500 件しか取得できない．
・空間検索の結果は，指定した空間範囲内のデータではあるが，位置情報を持つ Tweet は，GeotaggingAPI で投稿されたもののみであり，他の Tweet の位置情報は取得できない．

2.2 提案手法

不特定多数のユーザのデータを日付・場所ごとに収集するための手法を提案する．前節の方法(c)とは異なる方法で，Search API を利用する．以降で，取得したい日付を対象日，取得したいエリアを対象エリアと呼ぶ．また，Tweet ID を単に ID と呼ぶ．Tweet ID は，ユーザ ID とは別物であり，Twitter のサービス開始当初より，全 Tweet に対して投稿日時順に昇順の数値で付与されている ID である．Search API を単純に利用する場合，次の問題点がある．

・空間検索の結果は，指定した空間範囲内のデータではあるが，多くの Tweet について，個々の位置情報自体は取得できない(2.1 節)
・同一検索条件で 1,500 件までしか取得できない
・日付を指定して検索すると結果が大幅に間引

かれる

これらに対応するため，以下の手法を提案する．

・対象エリアできるだけ細かく分割し，それぞれのエリアでデータを取得

・Search API の検索条件に日付とページ(後述)を利用せず，代わりに ID を利用

具体的には，対象エリアを分割し，3 次メッシュ 4 メッシュ分(約 2km 四方)ごとにデータを収集する．この集計単位を，***3 次 2 倍メッシュ***と名付ける．メッシュコードは，左下の 3 次メッシュコードで代表する．図 5 の各メッシュが 3 次 2 倍メッシュである．

具体的な処理手続きを以降にまとめる．前提として，Search API は，1 回の API アクセスで，条件を満たす Tweet のうち，100 件のみ取得でき，100 件ずつを 1 ページとして，ページ番号を指定し以降の結果を取得するよう提供されている．

まず，分割した各エリアにおけるデータ収集処理(処理 B・後述)で利用するため，特定の 1 箇所の 3 次 2 倍メッシュを対象に，以下に示す日付境界 ID 取得処理(処理 A)を行う．対象日の翌日の Tweet の ID のうち，できるだけ小さな ID(対象日の翌日の 0:00 に近い時間の Tweet の ID)を取得する処理である．対象とする 3 次 2 倍メッシュは，日常的に Tweet 数が多いエリアが望ましい．

処理 A 日付境界 ID 取得処理

(Step1) 空間範囲と，対象日翌日の日付以前という条件を指定し API アクセスし，結果として，対象日翌日の Tweet を最新のものから，ID の降順に 100 件取得

(Step2) 処理 B(後述)を Step3 から開始

(Step3) 処理 B の終了後，最少の ID(結果の最後の Tweet の ID)を，求める ID とする．

この処理で取得した Tweet データ自体は破棄する．続いて，対象エリアを分割した全ての 3 次 2 倍メッシュにごとに，次のデータ収集処理を行う．

処理 B データ収集処理

(Step1) 日付境界 ID 取得で取得した ID を最大 ID として Step2 を実行

(Step2) 空間範囲と最大 ID を検索条件として指定し API アクセスし，結果として，指定した最大 ID より ID の小さい Tweet を，ID の降順に 100 件取得

(Step3) 得られた結果中に，対象日の前日の Tweet が含まれれば，それらを除外して収集処理を終了

(Step4) 得られた結果中，最少の ID(結果は ID 降順なので，うち最後の Tweet の ID)から 1 を引いた値を最大 ID として，Step2 に戻る

2.3 評価

提案手法と，2．1 節の一般的な手法(a)，(c)を用い，以下の条件を満たすデータを収集した．

空間範囲：3 次 2 倍メッシュ 53394600(有楽町駅付近中心約 2km 四方)

投稿日時の時間範囲：2011 年 8 月 20 日 0:00 より 24 時間分

結果を表 2 に示す．提案手法は，一般的な手法の 3 倍以上のデータ数を取得できた．提案手法では，次節に示す補間処理を 2 回行った．

表-2. 取得 Tweet 数

(a) Streaming API	31,711
(c) Search API	1,500
提案手法	97,787

3. 実装

3.1 データ収集システム

提案手法に基づくデータ収集システムを実装した．実装には，PHP と Perl を用いた．システム構成と処理内容は次のとおりである．

・日付境界 ID 取得用サーバ（1 台）

3 次 2 倍メッシュ 1 箇所に関して，トライアル収集(後述)と日付境界 ID 取得処理を行う

・収集サーバ（複数台）

分担する複数の 3 次 2 倍メッシュに関して，取得処理，複数回の補間処理(後述)を行う

日付境界 ID 取得用サーバは，収集サーバと同一マシンで問題ない．また，日付境界 ID 取得処理で収集した件数を，トライアル収集の件数とすることで，これらの処理は一つにまとめられる．

上記の構成と処理内容について補足する．Search API を利用するにあたり，次の実装上の課題があった．

(a) API のアクセス回数制限

API に同一の IP アドレスから短時間に大量にアクセスすると通信を遮断される．

(b) API の不安定性

提供されている API は，ベストエフォートのサービスであり，時折不安定になる．次節に示す収集

実験中も，1/10 程度に間引きした件数のみ返す期間が数日あった．また正常時でも，時折エラーを返すが，その際に同一条件で数回アクセスすると，正しく取得できる場合がある．

(a)について，次の対処を行った．

・分散環境での収集

複数の異なる IP アドレスからアクセスするように，複数のサーバで収集を行う．具体的には，各サーバが複数の 3 次 2 倍メッシュを分担して収集し，結果を集約する．

(b)について，次の対処を行った．

・トライアル収集

対象エリア全域の収集を行う前に，1 箇所の 3 次 2 倍メッシュで収集処理を行う．同じメッシュでのトライアル収集を継続的に行い，件数が極端に減った場合，エリア全域での収集処理開始を見合わせ，メールで警告を通知する．

・補間処理

指定した日付における，収集済みの全ての Tweet に関して，連続する 2 つの Tweet の投稿時刻に開きがある場合，大きい方の ID を最大 ID に指定し，小さい方の ID 以下の ID の Tweet を取得するまで，データ収集処理を行う

・API エラー時のリトライ

収集処理中に API エラーが発生した場合，指定した回数だけ同一条件で API アクセスする．

3.2　データ視覚化ツール

収集したデータを集計し時空間的な観点から視覚化するツールを開発した．具体的には，次を実現した．(1)は PHP と Perl，(2)(3)は adobe Flash/Flex で Google Maps API を利用し実装した．

(1) データ集計

指定した空間単位，時間間隔ごとに Tweet 件数を集計する

(2) メッシュ濃淡地図アニメーション生成

各メッシュの Tweet 数に基づく濃淡図のメッシュレイヤーを地図に重ねて表示する

(3) グラフ表示

地図上でユーザにより指定されたメッシュに関して，Tweet 数の時間変化をグラフ化する

4. データ収集実験

提案手法を用いたデータ収集実験を行った．サーバ 4 台で収集したが，分散環境での収集実験が目的であり，今回の結果の 2 倍程度の件数であれば，1 台のサーバで取得可能である．収集に要した期間は，収集したデータの時間範囲と同じく 2 週間である．その日の前々日 1 日分のデータ収集を，毎日行った．日付境界 ID 取得処理とトライアル収集処理には，3 次 2 倍メッシュ 53394600(有楽町駅付近中心)を利用した．API エラー時のリトライ回数は 3 回とした．各日のデータとも補完処理を 2 回行った．Search API へのアクセス頻度は，1 サーバ 1 時間あたり 300 アクセスとした．収集したデータの概要を表 3 に示す．

表-3.　収集したデータの概要

空間範囲	2 次メッシュで 533935 を左下とした 4 メッシュ分（都心部 約 20km 四方）
集計単位	3 次 2 倍メッシュ（約 2km 四方）
投稿日時の時間範囲	2 週間分（2011 年 7 月 25 日 0:00 より 1 週間と同 8 月 8 日より 1 週間）
tweet 数	3,476,059
ユーザ数	216,430

以降で，本データに関して，複数の観点から集計結果を示す．

4.1　日別変化

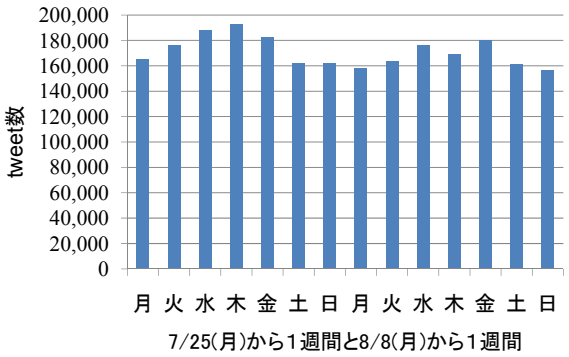

図-1.　Tweet 数の日別変化(対象エリア全域)

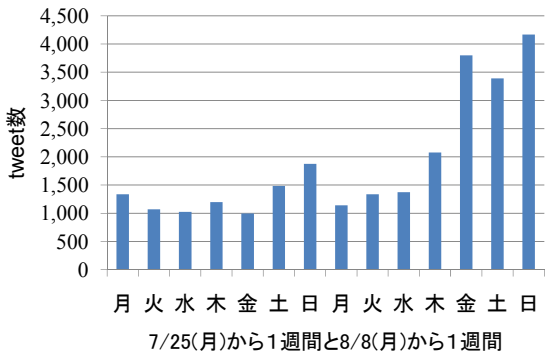

図-2.　Tweet 数の日別変化(お台場付近 2km 四方)

図 1 は，対象エリア全域の日別の件数変化を示

す．土日の件数が少ないこと分かる．平日の平均が 174,983 件，土日の平均が 160,369 件であり，約 9%の差がある．月曜日の件数も少ないが，日曜日の深夜(月曜日 0 時)に少ないことが影響している．図 3 のとおり各日とも件数のピークは 23 時から翌 0 時である．

一方で，特定のメッシュに着目すると，異なる傾向が見られる．図 2 は，3 次 2 倍メッシュ 53393642(お台場付近 2km 四方)の日別の件数変化を示す．週末に大幅に増加している．

4.2　時間変化

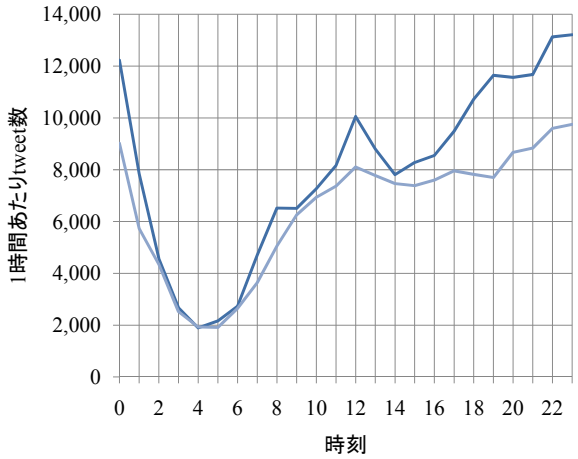

図-3.　Tweet 数の時間変化(対象エリア全域)

図 3 は，対象エリア全域の Tweet 数の時間変化を示す．日別の Tweet 数が最多の日(7/28(木))と最少の日(8/8(日))の 2 日分のグラフである．午前 4 時台が最少で，昼 12 時に一旦ピークがあり，翌 0 時のピークに向けて増加するという，大まかな傾向は共通している．

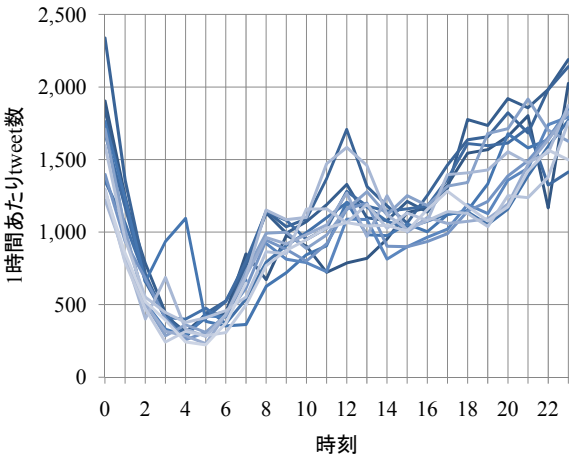

図-4.　Tweet 数の時間変化(有楽町駅付近中心約 2km 四方)

図 4 は，全期間にわたって日別の Tweet 数が最多だった 3 次 2 倍メッシュ 53394600(有楽町駅付近中心約 2km 四方)の Tweet 数の時間変化を示す．14 日分のグラフである．午前 4 時に普段と異なるピークがある日は 7/31 である．この時間，東京都心部で震度 3(福島県で震度 5 強)の地震があった．

図 5 は，3 次 2 倍メッシュの 1 時間あたりの Tweet 数の濃淡地図の時間変化を示す．日別の Tweet 数が最多の日(7/28(木))の様子である．開発した視覚化ツールは，このような時間変化をアニメーションで確認できる．

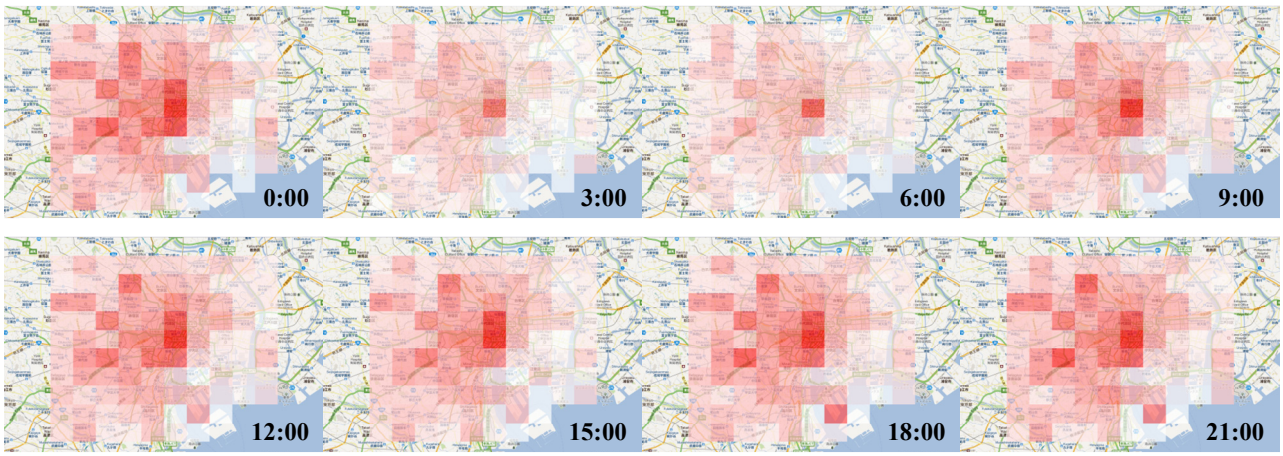

図-5.　1 時間あたりの Tweet 数の時間変化（Google Maps 利用．地図データ：ゼンリン）

4.3　ユーザ

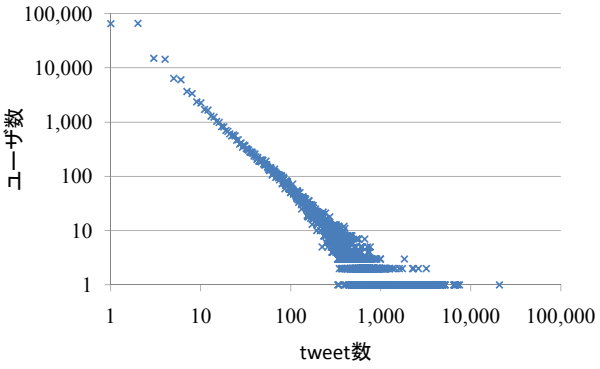

図-6.　ユーザあたりの Tweet 数とユーザ数

図 6 は，全期間中のユーザあたりの Tweet 数ごとに，ユーザ数を集計した結果である．Tweet 数が 4 件以下のユーザがほとんどを占める．Tweet 数最多のユーザ(2 週間で 20,744 件)を含め，極端に多いユーザは，スクリプトにより自動投稿を行うアカウント(ボット)である．

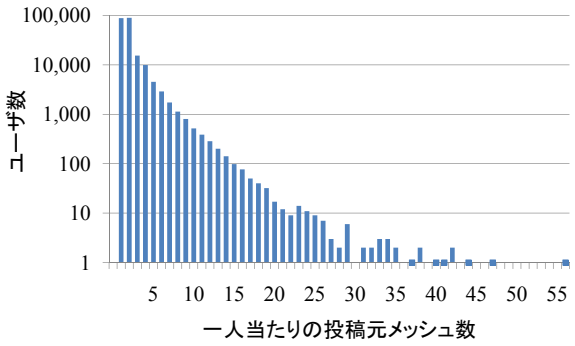

図-7.　ユーザあたりの投稿元メッシュ数とユーザ数

図 7 は，全期間中のユーザあたりの投稿元メッシュ数ごとに，ユーザ数を集計した結果である．半数以上のユーザが少なくとも 2 つ以上の異なるメッシュから投稿している．投稿元メッシュ数が最多のユーザは，2 週間で 56 の異なるメッシュから投稿している．

5. 結論と今後の課題

本論文では，Twitter のデータ収集システムを提案し，一般的な手法の数倍のデータ数の取得に成功した．また，収集したデータの時空間的な視覚化ツールを提案し，データの概要を示した．今後の課題として以下に取り組む予定である．

・データの収集と提供

システムをスケールアウトし，より広い範囲のデータを収集する．また，収集したデータを他の研究者に対して提供する枠組みを検討する．

・データの応用

得られたデータを利用し，ユーザ間のコミュニケーションと場所，特定の事象と場所，人の流れに着目した分析を行う

参考文献

Sakaki, T., Okazaki, M., and Matsuo Y., 2010. Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors. In Proc. of the 18th International World Wide Web Conference.

Fujisaka, T., Lee, R., and Sumiya, K., 2010. Exploring Urban Characteristics Using the Movement History of Mass Mobile Microbloggers. The Eleventh Workshop on Mobile Computing Systems and Applications.

van Liere, D., 2010. How far does a tweet travel?: Information brokers in the twitterverse. In Proc. of the International Workshop on Modeling Social Media.