

Twitterでのイベント検出を用いた電車混雑の予測システム Predicting System for Train Crowdedness using Event Detection on Twitter

岩淵 和紀[†] 萩原 威志[‡]
Kazuki Iwabuchi Takeshi Hagiwara

1 はじめに

近年、Twitter^[1]などのSNSやマイクロブログに着目して、研究に取り入れている事例が多くある。マイクロブログには、テキストを初めとする膨大な量の情報が溢れている。これらの情報を収集・分析することでさまざまな結果を得る研究が多々存在する。

SNS関係の研究の中には、SNSに投稿されたテキストを解析してイベントの発生を検出するという研究がある。これは、現在行われているイベントや将来予定されているイベントをテキストの特徴から見つけ出すといったものである。検出するイベントもさまざま、地方の祭りやコンサートなどの催し物から、電車の遅延や交通の渋滞などの出来事まで多岐に渡る。

ところで、そういったイベントが発生する日には電車などの交通機関が混むことが多いと感じる。たとえばサッカーの試合がある日には、試合の開始終了の前後はサポーターで電車が混んでいることがよくある。そういった混雑はイベントの種類や規模、開催場所や時刻によって、発生するかどうかやどれくらい混むかなどが決まると考えられる。また、人々の注目や関心の多さも関係していると思われる。そういった理由から、Twitterを用いて将来のイベントを検出し、そこから電車の混雑を予測することを目指す。

2 研究概要

本論文ではTwitterへの投稿を用いた、電車混雑の予測手法を示す。イベントが行われる日にはその参加者などが開催場所に集まる関係上、周囲の交通が混雑することが考えられる。特に電車網が発達して人口の多い都市部では地方に比べ、電車混雑の指標である混雑率が高い傾向にある^[2]。

そこでイベントの発生と電車混雑の相関を調べるため、イベントを検出しそこから電車の混雑を予測するシステムの構築を目指す。イベントの検出にはTwitterを用いた。これは、電車の混雑は人々のイベントに対する関心の強さにも依存しているという仮定のもと、Twitterは多くのユーザがさまざまなことをつぶやいているため、人々の現在の関心を反映していると考えられるからである。またイベントの情報を収集する際に、企業ごとに形式の異なるサイトよりも、140文字で簡潔に述べられているTwitterのほうが適していると考えたためである。

本研究で構築したシステムはクエリとして、日付時刻路線、方面を指定することで、イベントの発生を鑑みた電車の混雑度を出力として返し、ユーザが電車混雑の予測を知ることができるものである。

3 予測システムの概要

本システムは以下のサービスを利用している。

- Twitter REST API^[3]
- Google Maps API^[4]
- goo ラボ API^[5]
- Geocoding API^[6]
- こみれば及び電車混雑レポート^[7]

構築するシステムは以下の手順を踏まえることで、電車混雑の予測を実現する。

- ① Twitterへ投稿されたテキスト(以下、ツイート)を収集・分析し、イベントについて言及しているものとそれ以外のものに分類する。
- ② ツイートを、言及しているイベントごとにまとめる。
- ③ イベントごとにイベント情報をツイートから抽出する。
- ④ イベントごとのTwitterユーザの関心の高さ、評判の度合いをテキストから抽出する。
- ⑤ 電車混雑情報を「電車混雑レポート」から収集する。
- ⑥ ③と④を特徴量、⑤を教師値として分類器を学習させる。
- ⑦ Twitterから将来のイベントを検出し、分類器を用いて指定の日の混雑を予測する。

以下にそれぞれの手順の詳細を示す。

3.1 イベントの検出

ツイートの中からイベントについて言及しているツイートを分類する。まず、Twitter REST APIを用いてツイートを収集する。ツイートは「8月15日」、「来週の火曜日」などの日付表現の含まれたものを集める。これは、日付表現を含むツイートはその日にあるイベントについて言及していることがあるためである。次にツイートを形態素解析し単語に分けて、BoW(Bag of Words)の要領でベクトル化する。集めたツイートは手動でイベントについてのものでないものに分け、教師値を付加する。そして特徴ベクトルと教師値を学習データとして分類器に学習させる。分類器には二値分類の性能が高いSVM(Support Vector Machine)を用いた。学習したSVM

[†] 新潟大学大学院自然科学研究科

[‡] 新潟大学工学部情報工学科

を用いてツイートがイベントについてのものか否かを判別する。

3.2 イベントの同定

イベントツイートとして判別されたツイートを言及しているイベントごとにまとめる。まず任意の二つのツイートについて共通した単語のリストを求める。リスト内の各単語に設定された値の合計を求め、閾値以上であればその二つのツイートを同じイベントについてのものとして同定する。各単語の設定された値はその単語の出現頻度に反比例する値になっている。

3.3 イベント情報の抽出

検出したイベントについて、そのイベント情報を同定したツイート群から抽出する。イベント情報とは、イベントの開催期間、時刻、場所、ジャンルなどの開催情報のことを指している。同定した各イベントのツイート群について、WebAPIを利用してイベント情報を抽出する。GoogleMapsAPIとGeocodingAPIを用いてイベントの開催場所を、gooラボの固有表現抽出APIや時刻表現正規化APIを用いてイベントの開催期間、時刻、建物などを抽出した。

3.4 ユーザによる評判値の算出

Twitterを用いて、各イベントにユーザがどれくらい注目しているか、どのような関心を寄せているのかの度合いを算出する。これは、ユーザの関心の度合いやその評判によってイベントの来場者数に変動し、電車の混雑にも影響を与えたと考えられるからである。まず、各イベントツイートごとにテキストのネガポジ判定を行う。これは、テキストを単語に分けたのち、単語感情極性対応表¹⁾を利用してツイートの各単語に値を付加し、その平均値をツイートのネガポジ値とする。これは、イベントについてユーザが良く思っているか、悪く思っているかの指標となると考えられる。同イベントの全ツイートのネガポジ値の合計をそのイベントの評判値とする。また、各イベントについて言及しているツイートの数もユーザの関心の一つの指標とした。

3.5 電車混雑情報の収集

学習の際の教師値として、過去の電車の混雑情報を収集する。これは、株式会社ナビタイムジャパンが提供している「こみれば」及び「電車混雑レポート」を利用して電車混雑情報を収集する。「こみれば」とはユーザが現在の電車混雑情報を投稿し、共有することのできるサービスであり、「電車混雑レポート」で投稿された混雑情報を見ることが出来る。投稿されている情報は投稿時刻、路線、方面、駅、運行情報、混雑度(6段階)である。本研究ではこれらの情報を収集し、混雑度を教師値として分類器を学習させる。

3.6 分類器の作成・学習

集めた学習データを用いてイベント情報から混雑度を予測する分類器を作成する。イベント情報と評判値を学習の特徴量、特定の日の混雑度を教師値として分類器を学習させる。データには質的データと量的データが混在しているため、分類器にはランダムフォレストを用いた。

3.7 電車混雑の予測

学習させた分類器を用いて、電車混雑を予測する。まずTwitterとSVMを用いて将来予定されているイベントを検出し、そのイベント情報を抽出する。また、そのイベントについてのツイートを分析して、ユーザによる評判値を算出する。その後クエリとして、日付、時間帯、路線、方面を与えることで、得た情報と学習させた分類器を用いてその日時の電車の混雑度の予測をする。出力は指定した日時・路線・方面の電車の6段階で表される混雑度を返す。これにより、ユーザはどんなイベントが予定されているかを調べることなく、指定した日時の電車の混雑予測を見ることが出来る。

4 おわりに

本論文ではTwitterを利用した電車混雑の予測手法を示した。実際のシステムについては現在実装中であり、精度評価の実験と合わせて進めている。発表時にはシステムの動作結果および予測精度の評価結果を示す予定である。

また、本手法はTwitterを用いてユーザのイベントに対する関心の度合いを算出しているため、イベント情報のみを用いた混雑予測に比べ、イベントの来場者数など不確定な情報も考慮した予測に秀でている。そのため、過去に事例のないイベントが発生する場合でも混雑予測が可能であると考えられる。

今後の展望としては、イベント以外の要因による電車の混雑も考慮していく必要があると考えられる。要因としては、曜日や季節など特定の日時によるものや天候などが考えられる。解決策としては、年単位の学習データの収集や天気予報情報の抽出が挙げられる。これらを踏まえてより正確な予測システムを構築していきたい。

5 参考文献

- [1] Twitter
<http://twitter.com>
- [2] 混雑度データ,国土交通省
<http://www.mlit.go.jp/common/001025446.pdf>
- [3] Twitter REST API
<https://dev.twitter.com/rest/public>
- [4] Google Maps API
<https://developers.google.com/maps/>
- [5] goo ラボ
<https://labs.goo.ne.jp/>
- [6] Geocoding API
<http://www.geocoding.jp/api/>
- [7] 電車混雑レポート
<http://www.navitime.co.jp/?ctl=0171>
- [8] 単語感情極性対応表
http://www.lr.pi.titech.ac.jp/~takamura/pubs/pn_ja.dic