

大規模災害時における Twitter を用いた ネットワークシステム制御に有用な情報の抽出

丸 千尋[†] 榎 美紀^{†,‡} 中尾 彰宏^{‡,‡‡} 山本 周^{‡‡} 山口 実靖^{‡‡‡}
小口 正人[†]

[†] お茶の水女子大学 〒112-8610 東京都文京区大塚 2-1-1

[‡] 日本アイ・ビー・エム株式会社 〒135-8511 東京都江東区豊洲 5-6-52

^{‡‡} 東京大学 〒113-8654 東京都文京区本郷 7-3-1

^{‡‡‡} 工学院大学 〒163-8677 東京都新宿区西新宿 1-24-2

E-mail: †chihiro@ogl.is.ocha.ac.jp, oguchi@computer.org, ‡†enomiki@jp.ibm.com,

‡‡‡{nakao,shu}@iii.u-tokyo.ac.jp, ‡‡‡†sane@cc.kogakuin.ac.jp

あらまし 地震など大規模災害時には、電話やインターネットが繋がりにくくなるといった問題が発生する。現段階では障害の検知と判断を手で行っているが、災害時のようにデータの種類や量が桁違いに増えると、少数のネットワーク管理者が多数の障害を手作業で検知するのに限界がある。そこで本研究は、SNS による集合知を利用した障害検知を迅速かつ高精度に行うシステムを提案をする。集合知による障害検知では、ツイートの「位置情報による」検知と「時間変化のフィルターによる」検知を行うことで、ネットワーク障害検知の精度を上げることができる。障害検知の自動化では、Twitter から検出したネットワーク障害に関する情報を出力することで、ネットワーク制御システムに直接連携するために、ネットワーク障害検知の「自動化システム」の構築を行う。2014 年 11 月 22 日に長野県北部で発生した地震を対象に自動化システムの動作を確認すると、ネットワーク障害に関する第一報が企業から出される前に、ネットワーク障害を検出できるという結果が得られる。このことは、提案手法は SNS による集合知を利用した精度の高い障害検出を実現可能であることを示している。

キーワード Twitter, イベント検知, 大規模災害, ネットワーク障害, ネットワークシステム制御

Information Detection on Twitter for Network System Control in Large-Scale Disasters

Chihiro MARU[†], Miki ENOKI^{†,‡}, Akihiro NAKANO^{‡‡}, Shu YAMAMOTO^{‡‡‡}, Saneyasu
YAMAGUCHI^{‡‡‡‡}, and Masato OGUCHI[†]

[†] Ochanomizu University 2-1-1 Otsuka, Bunkyo-ku, Tokyo, 112-8610, Japan

[‡] IBM Research - Tokyo 5-6-52 Toyosu, Koto-ku, Tokyo, 135-8511, Japan

^{‡‡} Tokyo University 7-3-1 Hongo, Bunkyo-ku, Tokyo, 113-8654, Japan

^{‡‡‡‡} Kogakuin University 1-24-2 Nishi-shinjuku, Shinjuku-ku, Tokyo, 163-8677, Japan

E-mail: †chihiro@ogl.is.ocha.ac.jp, oguchi@computer.org, ‡†enomiki@jp.ibm.com,

‡‡‡‡{nakao,shu}@iii.u-tokyo.ac.jp, ‡‡‡†sane@cc.kogakuin.ac.jp

1. はじめに

地震や台風などの大規模災害が発生すると多数のユーザーが同時にネットワークを利用することなどにより、電話やインターネットが繋がりにくくなるといった問題が生じる。このような

緊急時に、電話やインターネットが利用可能であることは重要である。従って、大規模災害時における障害検知の必要性は高い。現在、人間が見て判断可能な単純なネットワーク制御は行われている。しかし、大規模災害時のようにデータの種類や量が桁違いに多くなると、少数のネットワーク管理者が多数の障

害を手作業で検知するのに限界がある。

この問題を解決するために、本研究は、ソーシャルネットワークサービス (SNS) による集合知を利用した障害検知を迅速かつ高精度に行うシステムを提案する。マイクロブログサービスの一つである Twitter [1] は人々の生活の一部として定着しており、Twitter から得ることのできるソーシャル情報は実世界の動きを反映していると言うことができる。そこで、本研究では、Twitter を用いた障害検知を行う。

本論文の貢献は以下の 4 つである。

- (1) SNS を利用して「集合知による」障害検知を行う。
- (2) ツイートの「位置情報による」検知の精度を向上する。
- (3) ツイートの「時間変化のフィルターによる」検知の精度を向上する。

- (4) 検知の「自動化システム」を提案し、動作を確認する。

本稿の構成は以下のとおりである。2. 章で関連研究について述べ、3. 章で提案システムの概要を紹介し、4. 章で提案システムを候補データ抽出処理と外部情報マッチング処理に分け、詳細に紹介する。5. 章で候補データ抽出処理により取得した候補データの評価を行い、6. 章で提案手法を用いて検知した障害と 4. 章で取得した外部情報の可視化を行い、提案システムの有効性を確認する。7. 章で提案システムの自動化を行い、Twitter の実際のデータを用いて自動化システムの動作を確認する。最後に、8. 章で本稿をまとめる。

2. 関連研究

SNS でのデータを解析することで、地震や土砂災害や火災など、実世界で発生する出来事を検知する手法が数多く存在する [2] [3] [4] [5] [6]。Sakaki ら [2] は、Twitter のユーザをソーシャルセンサとすることで、地震を早期に発見し、発生場所の推測をする手法を提案している。国土交通省国土技術政策総合研究所 [3] は、災害が発生する可能性のある地域の住民等のツイートを元に、土砂災害の予兆や発生の早期把握を行っている。これらの研究は、出来事の発生の有無に焦点を当てており、大きな出来事が発生した際のより詳細な情報の抽出はなされていないため、大規模災害が原因で起こるネットワーク障害を検知することを目標としている本研究とは異なる。また、これらの研究は、公式な機関から発行される外部情報を使わずに、ソーシャル情報のみを解析して特定の出来事の検知を行っている点において、外部情報も同時に取得して精度を上げている本研究とは異なる。その一方、本研究は、ツイートの「位置情報による」検知と「時間変化のフィルターによる」検知を行うことで、集合知によるネットワーク障害検知の精度を上げ、ネットワーク管理に活かすことのできる情報の抽出を行っている点で新規性がある。

また、我々はネットワーク管理を行うために、SNS を利用する。Tongqing [7] らは、モバイルネットワーク機能に問題があった場合、ユーザはサービスセンタに電話するよりも早く Twitter に問題を投稿することを示した。Takeshita ら [8] [9] は、ネットワークの故障状況を把握するために、Twitter 上に投稿されたネットワーク問題に言及したツイートを利用して

いる点で本研究と動機が似ている。しかしながら、Takeshita らは自然災害を対象にしていない点で本研究と異なる。また、Twitter のメッセージの情報のみを用いて障害を判断している点で本研究と異なる。

3. 提案システムの概要

本研究では以下の「ネットワーク障害検知システム」を提案する。提案システムの概要を図 1 に示す。

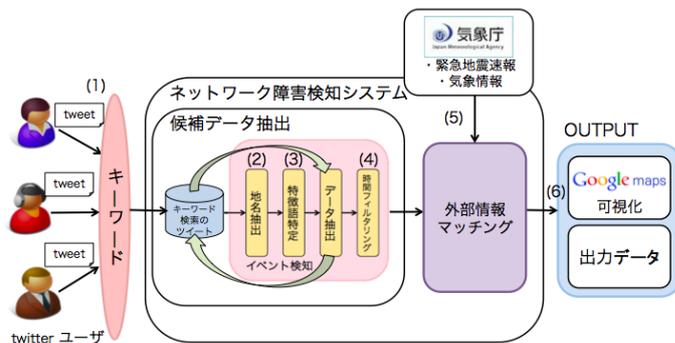


図 1 ネットワーク障害検知システム

- (1) 特定のキーワードを設定して、キーワードを含むツイートを取得する。
- (2) キーワード検索で取得したツイートを同じ地名名詞が含まれるツイートごとにまとめる。
- (3) 地名名詞ごとにまとめたツイートの中から特徴語を特定し、特徴語を含み地名名詞を含まないツイートを、キーワード検索で取得したツイートの中から抽出して加える。
- (4) 関係のないツイートを排除するために、ツイートされた時間を考慮し、時間フィルタリングを行う。
- (5) 緊急地震速報や気象情報などの外部情報を取得し、解析を行う。
- (6) ツイートから検知した障害と外部情報を照らし合わせ、Google マップ上に可視化する。

また、Twitter から検出したネットワーク障害に関する情報を出力することで、ネットワーク制御システムに直接連携するという目的を達成するために、ネットワーク障害検知システムの自動化を行う。自動化にあたり、キーワード検索で取得したツイートのサンプリング回数を 1 分とし、現在のツイートから 60 分遡った時刻までのツイートを障害検知対象のツイートとして扱う。この自動化されたシステムにより、外部データと集めたツイートを障害場所ごとにまとめたものが、出力データとして出力される。

4. システム詳細

4.1 SNS からの候補データの抽出

4.1.1 キーワード検索

Twitter 社の Search API [10] を使って取得したツイートを、特定のキーワードを用いてキーワード検索を行う。キーワードは、「電話 and 繋がらない」や「ネット and 繋がらない」など、ネットワーク障害が起きていることが分かるものに設定する。

4.1.2 地名抽出

キーワード検索で取得したツイートの本文とユーザのプロファイルと GEO タグを MeCab [11] を使ってそれぞれ品詞分解し、地名名詞を抽出する。GEO タグの緯度・経度は、Yahoo!社が提供する Yahoo!リバースジオコード API [12] を使って市区町村に変換する。抽出した地名名詞の出現回数をカウントし、一定回数以上出現した地名名詞ごとに、キーワード検索で取得したツイートをまとめる。ここでは出現回数の閾値を5回とする。

4.1.3 特徴語特定

地名名詞ごとにまとめたツイートだけでは、同じ障害について言及したツイートであるが地名名詞を含んでいないツイートを全て捨ててしまうことになる。そのため、地名名詞ごとにまとめたツイートの中に出現する特徴的な単語を抽出し、抽出した特徴語を含み地名名詞を含まないツイートを、キーワード検索で取得したツイートの中から加えることを考える。

特徴語は、地名名詞ごとにまとめたツイートの中から、MeCab を使って名詞のみを抽出することで見つける。その際、ノイズを排除するために、正規表現を用いて半角記号と半角数字は取り除く。抽出した名詞それぞれに対して tf-idf 値を求める。そして、tf-idf 値が 0.2 以上となった名詞を特徴語と定義する。tf 値と idf 値の計算式を以下に示す。ここで、抽出した名詞を N 、実験対象ツイートを TW_e 、Twitter 社が提供する Streaming API [13] を利用して取得した約 10 万件のツイートを TW_{10} とする。 TW_{10} は、idf 値を計算する際に用いる。

$$tf \text{ 値} = \frac{TW_e \text{ 中に } N \text{ が出現する回数}}{TW_e \text{ 中に出現する総単語数}}$$

$$idf \text{ 値} = \frac{TW_e \text{ と } TW_{10} \text{ のツイート数の和}}{TW_e \text{ と } TW_{10} \text{ の中で } N \text{ を含む数}}$$

2014 年 7 月 8 日にキーワードを「電話 and 繋がらない」としたときに取得したデータに対して、特徴語の特定を行う。結果を表 1 に示す。

表 1 特徴語を抽出した結果

(a) 北海道		(b) 札幌		(c) 苫小牧	
関連語	tf-idf 値	関連語	tf-idf 値	関連語	tf-idf 値
地震	0.36	地震	0.28	地震	0.35
		番号	0.25	震源	0.29
		iPhone	0.21	ビックリ	0.24
		心配	0.21	大丈夫	0.21

結果より、地震に関係のある名詞を特徴語として抽出できることがわかる。また、地震が発生したときに「地震」「震源」などの名詞が特徴語として抽出されるのは予想することができるが、「心配」「ビックリ」「大丈夫」といった感情的・感覚的な名詞も抽出することができる。このことは、有益そうな特徴語を発見することが可能であるということを示している。

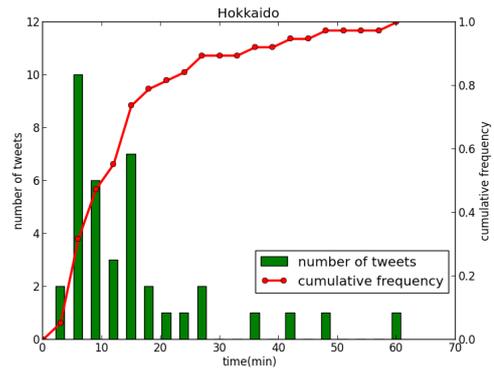
4.1.4 時間フィルタリング

地名名詞でまとめたツイートの中には、内容の異なるツイートが多く混じっている。そのため、ツイートされた時刻を考慮

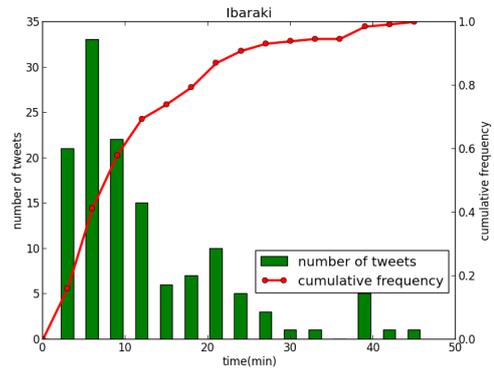
して関係のないツイートを排除することを考える。大規模災害が発生した際には、Twitter 上の複数のユーザが特定の時間に似たような内容のツイートをする [2]。本研究では、その点に着目して、ツイートを排除する時間の閾値を決定する。閾値を決定するために、ネットワーク障害に言及するツイート数の時間変化を調べ、一般化する。

• 地震の場合

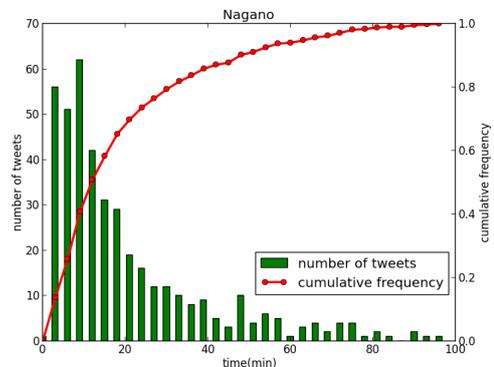
2014 年 7 月 8 日に北海道で発生した地震、2014 年 9 月 16 日に茨城県南部で発生した地震、2014 年 11 月 22 日に長野県北部で発生した地震の際のネットワーク障害等に言及するツイート数の時間変化を図 2 の緑の棒グラフに示す。



(a) 北海道での地震



(b) 茨城での地震



(c) 長野での地震

図 2 ネットワーク障害に言及するツイート数の時間変化 (地震の場合)

図 2 より、地震が発生した直後にツイート数が急速に増えて、

その後収束していることがわかるので、地震の場合はツイート数の時間変化が特徴的な形になると考えられる。本研究ではサンプル数が少ないため、累積度数を考える。累積度数の実データを図2の赤線で示す。累積度数の時間変化を見ると、指数分布の累積分布関数に似た形になっている。そのため、指数分布の累積分布関数を用意し、最小二乗法を使ってフィッティングを行う。指数分布の累積分布関数を以下に示す。

$$f(x) = 1 - e^{-\lambda x}$$

北海道、茨城、長野で発生した地震をフィッティングした結果を図3に示す。パラメータ λ はそれぞれ0.07372231, 0.09200698, 0.0536034とした。

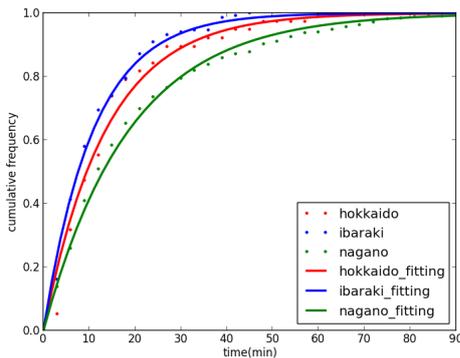


図3 指数分布の累積分布関数にフィッティングした結果

どの実験対象ツイートも指数分布の累積分布関数にフィッティングさせることができるという結果が得られた。このことから、地震の場合のツイート数の時間変化は指数分布で近似ができることがわかる。また、指数分布であることから、閾値が60分で8割の事象を捉えることができるという結論が得られる。

● 台風の場合

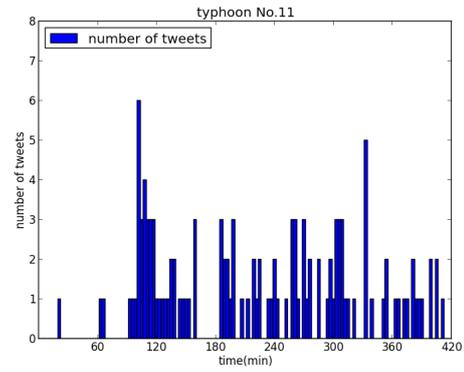
2014年7月29日に発生した台風11号と2014年10月3日に発生した台風19号の際のネットワーク障害等に言及するツイート数の時間変化を図4の青の棒グラフに示す。

図4より、台風の際は、図4(a)のように少ないツイート数が長く続いている場合と、図4(b)のように急にツイート数が増える場合の2種類に分類できることがわかる。地震の場合と同様に、図4(b)の台風19号のように特徴が見えやすいケースの累積度数を考えることにする。累積度数の実データを図4(b)の赤線で示す。累積度数の時間変化を見ると、ロジスティック分布の累積分布関数に似た形になっている。そのため、ロジスティック分布の累積分布関数を用意し、最小二乗法を使ってフィッティングを行う。ロジスティック分布の累積分布関数を以下に示す。

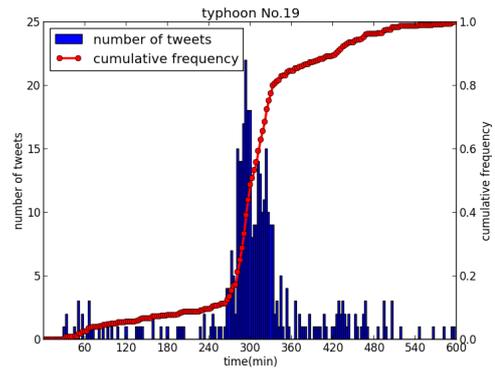
$$f(x) = \frac{1}{1 + e^{-(x-\mu)/s}}$$

μ は平均である。台風19号をフィッティングした結果を図5に示す。

結果より、台風の場合は地震の場合と比べて、綺麗にフィッティングさせることができないことがわかる。これは、地震が起こるのは一瞬なのに対し、台風は継続的に被害が続くためだ



(a) 台風11号



(b) 台風19号

図4 ネットワーク障害に言及するツイート数の時間変化(台風の場合)

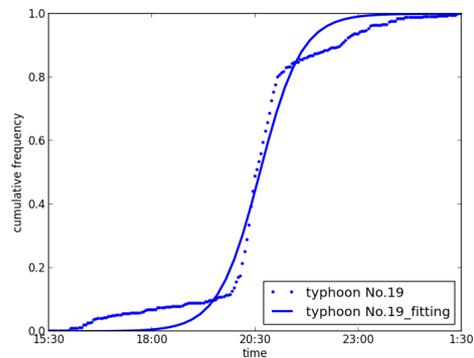


図5 ロジスティック分布の累積分布関数にフィッティングした結果

と考えられる。このことは、台風の場合のツイート数の時間変化は特徴があまり明確でないことを示している。

4.2 外部情報のマッチング

本システムで検出された障害の原因や場所を特定するために、公的機関から発行される災害情報を取得してマッチングを行う。

4.2.1 外部情報の取得

Twitterの情報は即時性があるが、必ずしも正確であるとは言えない。そのため公的機関から発行される外部情報を取得し、信頼性を確保する。本研究ではTwitter経由で発信される緊急地震速報や気象情報の公式ツイートを外部情報として利用する。

緊急地震速報は、法人が財団法人気象業務センターと契約することで受信ができる。そのため本研究では、Twitter社が提

供する Streaming API を使って、Twitter 上で発表される緊急地震速報のツイートを検析して間接的に利用し、地震の発生時刻や震源地やマグニチュードを取得する。

気象情報も同様に、Twitter 上で発表される気象情報のツイートを検析して間接的に利用し、台風情報や大雪情報を取得する。

4.2.2 外部情報とのマッチング

得られた外部情報と本システムで検出された候補ツイートとのマッチングを行う。実際に障害が発生している場所と、障害は起きていないがツイートされた場所はそれぞれ異なる可能性がある。例えば、北海道で地震が発生した場合、北海道から東京に電話かけたい人と、東京から北海道に電話かけたい人がいたとすると、どちらも電話が繋がらないという内容のツイートをするが、実際は地震が原因で北海道の電話網に障害が発生している、ということ特定したい。実際に障害が発生した場所は、災害が発生した場所に近いと考え、外部情報の発生時刻と位置の情報を用いてマッチングを行う。

例えば地震の場合、障害が発生している地域は、震源地付近であると考えられる。そこで、緊急地震速報から取得した地震発生位置(緯度、経度、震源の深さ)と地震規模(マグニチュード)を用いてそれぞれの地域の推定震度の算出を行う。その結果、予測震度が4以上になった地域を障害が発生している場所、それ以外を障害は起きていないがツイートされた場所と決定する。予測震度は、防災科学技術研究所の地表面における最大速度と計測震度の関係式を用いて算出する[14]。関係式を以下に示す。

$$I_{INSTR} = 2.68 + 1.72 \log PGV_s$$

I_{INSTR} は計測震度、 PGV_s は地表面における最大速度 (cm/s) を示す。

5. 実 験

キーワード検索で取得した Twitter のデータに対して、候補データ抽出処理を行って抽出される候補データについて評価する。

5.1 実験データ

2014年7月8日、9月16日、11月22日にキーワードを「電話 and 繋がらない」としたときに取得した、北海道、茨城、長野で地震が発生した際のデータを用いる。

5.2 実験シナリオ

キーワード検索で取得したツイートに対して、三種類の手法を適用してそれぞれ候補データを抽出する。三種類の手法は以下の通りである。

- (1) 地名抽出
- (2) 地名抽出 + 関連キーワード抽出
- (3) 地名抽出 + 関連キーワード抽出 + 時間抽出

検証方法としては、まず、キーワード検索で取得したツイートから正解のツイートを人手で集める。正解のツイートとは、地震が原因でネットワーク障害が起きていることが明らかに分かるツイートである。そして、正解のツイートと、抽出した候

補データそれぞれとの適合率、再現率、F値を求める。ここで、適合率は、候補データの中にどの程度正解のツイートが含まれているかを示す。再現率は、正解のツイートの中に候補データに存在する正解のツイートがどの程度含まれているかを示す。F値はその両者の調和平均である。

5.3 実験結果

キーワード検索で取得したツイートに対して、三種類の手法を適用したときの結果を表2に示す。

表2 抽出された候補データの評価
(a) 北海道

手法	適合率	再現率	F 値
(1)	0.64	0.7272	0.6808
(2)	0.5714	0.9090	0.7017
(3)	0.9	0.8181	0.8570

(b) 茨城

手法	適合率	再現率	F 値
(1)	0.5918	0.4264	0.4956
(2)	0.5714	0.8235	0.6746
(3)	0.875	0.8235	0.8484

(c) 長野

手法	適合率	再現率	F 値
(1)	0.8071	0.6141	0.6974
(2)	0.6748	0.8913	0.7680
(3)	0.9625	0.8369	0.8953

適合率は特徴語を加えると下がってしまうが、その後ツイートされた時刻を考慮しノイズとなっているツイートを排除することで上げることができる。再現率は時間を考慮すると少し下がってしまうが、特徴語を含むツイートを加えることで大幅に上げることができる。また、手法(1)、(2)、(3)を試すにつれてF値を徐々に上げることができ、三種類の手法を全て使うことによって最終的にどの実験ツイートもF値を0.85付近にすることが可能である。

6. 検知した障害の可視化出力

提案手法を使って Twitter から検知した障害と緊急地震速報から取得した情報とのマッチングを行い、Google マップ上に表示することで、提案システムの有効性を確認する。2014年9月16日に茨城県南部で発生した地震と2014年11月22日に長野県北部で発生した地震の結果を図6と図7に示す。

ピンの色はツイート数を表しており、多い方から赤桃黄緑青の順である。ツイート数は、赤のピンが101以上、桃のピンが51以上100以下、黄のピンが21以上50以下、緑のピンが11以上20以下、青のピンが5以上10以下である。一方、外部情報から特定した震源地を円の中心に設定し、マグニチュードに応じて円の半径が決まる。ピンをクリックすると、図6(b)と7(b)のようにピンが刺さっている地域に関するツイートが出力される。

図6(a)と7(a)を見ると、円に近い場所ほど多くツイートが

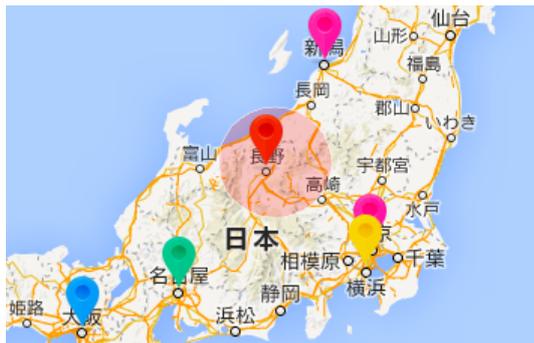


(a) 可視化の結果 (1)



(b) 可視化の結果 (2)

図 6 茨城県南部で発生した地震を可視化した結果



(a) 可視化の結果 (1)



(b) 可視化の結果 (2)

図 7 長野県北部で発生した地震を可視化した結果

検出されており、震源地付近で大きな障害が起きていることがわかる。このことは、障害の原因は地震であり、ピンの色と円の位置から提案手法を使って取得した障害と外部情報が一致していることを示している。

また、可視化の結果から、実際に障害が発生している場所と、

障害は起きていないがツイートされた場所はそれぞれ異なっているので、4.2.2 節の内容が必要であることがわかる。

7. ネットワーク障害検知システムの自動化

Twitter から検出したネットワーク障害に関する情報を出力することで、ネットワーク制御システムに直接連携するという目的を達成するために、本研究ではネットワーク障害検知の「自動化システム」を構築する。自動化にあたり、キーワード検索で取得したツイートのサンプリング間隔を 1 分とし、現在の時刻から 60 分遡った時刻までのツイートを障害検知対象のツイートとして扱い、提案手法を適用する。

2014 年 7 月 8 日 18 時 05 分に北海道で発生した地震と 2014 年 9 月 16 日 12 時 35 分に茨城県南部で発生した地震と 2014 年 11 月 22 日 22 時 08 分に長野県北部で発生した地震と 2015 年 2 月 6 日 10 時 25 分に徳島県で発生した地震を対象に、キーワードを「電話 and 繋がらない」とした Twitter のデータを用いて自動化システムの動作を確認する。ネットワーク障害に関する第一報が企業から出された時刻と自動化システムが障害を検出した時刻の結果を表 3 に示す。

表 3 自動化システムが障害を検出した時刻との比較

震源地	発生時刻	第一報時刻	検出時刻
北海道	18:05	なし	18:17
茨城	12:35	なし	12:37
長野	22:08	22:56	22:13
徳島	10:25	なし	10:42

表 3 より、長野で発生した地震の場合、企業がネットワーク障害に関する第一報をホームページに掲載した時刻が、地震発生から 48 分後の 22 時 56 分であるのに対し、自動化システムを用いて障害検知を行うと、地震発生から 5 分後の 22 時 13 分に障害を検出することができる。このことから、自動化システムを用いることで、第一報よりも大幅に早く障害を検出することができることがわかる。他の地震に関しては、北海道で発生した地震は地震発生から 12 分後、茨城県で発生した地震は地震発生から 2 分後、徳島県で発生した地震は地震発生から 17 分後に障害を検出する。茨城や長野で発生した地震に比べて、北海道や徳島で発生した地震の方が障害を検出するのに少し時間がかかるのは、そこに居住する人口が少ないからであると考えられる。しかし、どの地震に関しても、地震発生後、ネットワーク障害を短時間で検出することを実現できることがわかる。このことは、自動化システムの有効性を示している。

表 4 に、長野県北部で発生した地震の際に自動化システムで取得した地名名詞の時間変化を示す。

結果より、長野、新潟など震源地に近い地域から順に抽出されていることがわかる。東京や神奈川や千葉が抽出されるのは、実際にそこで問題が生じているのではなく、そこに居住しているユーザが震源地付近に電話をかけようとした際に、電話が繋がらないためである。表 4 より、地震発生から約 1 時間後から抽出される地名名詞が徐々に減っていき、最終的に 2 時間 30 分程度で収束することがわかる。

表 4 地名名詞の時間変化

時刻	地名名詞
22:08	
22:13	長野
22:14	長野 新潟 東京
22:23	長野 新潟 東京 神奈川
22:34	長野 新潟 東京 神奈川 千葉
23:17	長野 新潟 東京 神奈川
23:18	長野 東京
00:04	長野
00:26	
00:35	長野
00:46	

8. まとめと今後の課題

少数のネットワーク管理者による障害検知では、大規模災害時の多数の障害を手作業で検知するのに限界がある。この問題を解決するために、本研究は集合知を利用する自動化された障害検知を容易かつ高精度に行うシステムの提案を行った。集合知による障害検知では、「位置情報による」検知と「時間変化のフィルターによる」検知を行うことで、ネットワーク障害検知の精度を上げることができた。また、評価を行うことで、候補データ抽出処理の効果を確認した。障害検知の自動化では、Twitter から得られたネットワーク障害に関する情報を出力することで、ネットワーク制御システムに直接繋げるために、ネットワーク障害検知システムの自動化を行った。2014 年に北海道、茨城、長野で発生した地震と 2015 年に徳島で発生した地震を対象にして、自動化システムの動作を確認し、地震発生後、短時間で障害を自動的に検出できることを示した。これにより、自動化システムの有効性を示すことができた。また、Twitter の情報は必ずしも正確ではないという問題を解決するために、緊急地震速報などの公式の外部情報を取得して信頼性を高めた。そして、提案システムから検知した障害と緊急地震速報から取得した情報を照らし合わせ、Google マップ上に可視化を行った。

本稿では、地震が発生した場合の障害検知の自動化システムを構築したが、今後は地震以外の様々な大きな出来事にも対応できるように、自動化システムの拡張を行いたい。また、東京などの大都市が、障害検出を行う度に抽出されてしまったことも今後の課題である。これは、大都市に居住している人ほど Twitter を利用している割合が高いためだと考えられ、検出された地名に住む人口を加味して障害検知を行うことで解決できると考えられる。

謝 辞

本研究は一部、総務省戦略的情報通信研究開発推進事業 (SCOPE) 先進的通信アプリケーション開発推進型研究開発によるものである。

文 献

- [1] Twitter, <http://twitter.com/>
- [2] Sakaki, Takeshi, Makoto Okazaki, and Yutaka Matsuo. "Earthquake shakes Twitter users: real-time event detection by social sensors." Proceedings of the 19th international conference on World wide web. ACM, pp. 851-860, 2010.
- [3] 国土交通省: ソーシャルメディア分析によるリアルタイム災害発生情報検知手法の確立に関する研究, 2014
- [4] 斎藤翔太, 伊川洋平, and 鈴木秀幸. "Twitter を用いた災害情報の早期発見 (言語理解とコミュニケーション)." 電子情報通信学会技術研究報告= IEICE technical report: 信学技報 114.81 (2014): 7-12.
- [5] Sadilek, Adam, Henry A. Kautz, and Vincent Silenzio. "Predicting Disease Transmission from Geo-Tagged Micro-Blog Data." AAAI. pp.136-142, 2012.
- [6] Metaxas, Panagiotis Takis, Eni Mustafaraj, and Daniel Gayo-Avello. "How (not) to predict elections." Privacy, security, risk and trust (PASSAT), 2011 IEEE third international conference on and 2011 IEEE third international conference on http://social computing (SocialCom). IEEE, pp. 165-171, 2011.
- [7] Qiu, Tongqing, et al. "Listen to me if you can: tracking user experience of mobile network on social media." Proceedings of the 10th ACM SIGCOMM conference on Internet measurement. ACM, pp. 288-293, 2010.
- [8] 竹下恵, 横田将裕, 西松研, 長谷川治久, "SNS データを用いた NW 故障情報の抽出法," 電子情報通信学会 2012 年ソサイエティ大会, B-7-35, Sep. 2012.
- [9] 竹下恵, 横田将裕, 西松研, 長谷川治久, "SNS データを用いた NW 故障情報の抽出法の評価," 電子情報通信学会 2013 総合大会, B-7-44, Mar. 2013.
- [10] Twitter Serch API, <https://dev.twitter.com/rest/public/search>
- [11] MeCab, <http://mecab.sourceforge.net/>
- [12] Yahoo!リバースジオコード API, <http://developer.yahoo.co.jp/webapi/map/openlocalplatform/v1/reversegeocoder.html>
- [13] Twitter Streaming API, <https://dev.twitter.com/streaming/overview>
- [14] 独立行政法人防災科学技術研究所, "地震動の評価モデル", http://www.j-map.bosai.go.jp/j-map/result/tn_275/html/html/2-7.html