

### ツイッターにおける関連発言抽出手法の提案

MARUYA, Takuma / 丸谷, 琢磨

---

(出版者 / Publisher)

法政大学大学院情報科学研究科

(雑誌名 / Journal or Publication Title)

法政大学大学院紀要. 情報科学研究科編 / 法政大学大学院紀要. 情報科学研究科編

(巻 / Volume)

9

(開始ページ / Start Page)

155

(終了ページ / End Page)

160

(発行年 / Year)

2014-03

(URL)

<https://doi.org/10.15002/00010538>

# ツイッターにおける関連発言抽出手法の提案

## Study on Extract a Related Tweets with Influenced Tweet on Twitter

丸谷 琢磨

Takuma Maruya

法政大学大学院情報科学研究科情報科学専攻

E-mail: takuma.maruya.4w@stu.hosei.ac.jp

### Abstract

*Twitter has a powerful communication tool. Everyone can omit their information as they like and people who read it can spread it to other people by retweet as well. But there is no information about the reason why they retweet the tweet. Therefore the people can deliver misinformation and it is difficult to get related tweets. It makes misinform to twitter users. This paper proposes a system to extract related tweets associated with a certain tweet, which is retweeted by many people. An existing way, which Twitter supports to get related tweets by JSON, is not effective especially at coverage of related tweets. This research made a test collection including 4000 tweets from 1000 people about a certain tweet, which includes each posted 2 tweets after and before user retweets the certain tweet. To this test collection,  $f$ -measure and recall of the existing way have the scores 0.0047 and 0.23%, respectively. The proposed system runs for the test collection. The experiment shows that bag-of-words with feature of only noun and without wordnet has a highest  $f$ -measure. It scores 0.811 and at the time recall scores 0.776, which is 165 times than the existing way. This research is possible to apply this system for marketing of company and preventing to spread misinformation.*

### 1. 序説

近年、140文字以下の文章であるツイートを投稿し、ユーザー同士で交流を行うマイクロブログのTwitter[1]が流行している。米Twitter社のSEC提出文書によると2013年6月時点において、2億1830万人との報告がある[2]。これはアクティブユーザー数と呼ばれる1ヶ月以内に1度以上ログインしたユーザーの数(MAU)ではあるが、非常に大勢の人間が利用していることが分かる。利用者が多いためTwitterを用いたユーザークラスタリング、トピック抽出といったコミュニケーション支援の研究やソーシャルメディア・マーケティング、政治の世論に関する調査、感情・評判抽出などコミュニケーションから派生した内容の活用が盛んに行われている。個人単位では情報取得ツールの一旦を担いながら、情報発信ツールとしての役割も果たし、有益なサービスとして認知されている。また企業単位ではユーザー情報や投稿文の分析・企業自らの情報発信を行い、商業利用を図っている。

Twitterは平常時のみならず被災時のような緊急時にも有効である。日本では2011年3月に発生した東日本大震災後からTwitterの有用性が認められ研究が盛んになったが[3][6][7]、海外では2010年2月に発生したチリ地震などを決起に研究が行われていた[4]。Twitterは震災時のような非常時の情報取得手段として有効[5]な一方で、同時にデマ情報の情報拡散も行われてしまい、社会での混乱が発生した。上島ら[7]はデマと関連した概念として噂・風評・流言を挙げているが、Twitterにおける噂の研究としてはMendozaらの研究[4]がある。Mendozaらの研究の中でニュースと噂の対比を行い、噂は懐疑を抱いた状態で広がる傾向があるため、テキスト内容を解析すれば噂を検知することが出来るのではないかと示唆している。

またデマ拡散の原因の一端として情報拡散されたツイートの一人歩きが挙げられる。これはユーザーが情報拡散を行った際に、肯定や否定、懐疑など、どのような意図で情報拡散を行ったかシステムの仕様上、不明なことが原因である。そのため拡散されてきたテキスト情報のみしか判断材料がなく、衝撃や他者に伝えてあげたいと感じた場合、冷静さを欠いたまま情報拡散してしまう。またTwitter上に投稿された文章の病気関連情報をWeb上から抽出する手法[8]も考案されているが、Twitterに投稿された文章内容ではなく、文章自体に言及した関連情報を抽出することが出来ない。

そこで本論文では大多数の人間によって情報拡散されたツイートに対し、ユーザーが視野を広げた多面的解釈が出来るようにするため、拡散されたツイートに関連したツイートの抽出を行う。これはシステム上の関連発言検索機能ではカバレッジが低いためである。この研究は企業のマーケティングにおいて商品やイベント告知を行い大人数に情報拡散された場合、どのような意図で反響があったのか調べることを可能にし、デマ情報だけでなくマーケティングにも応用可能である。

以降、第二章では実験で使用する関連手法について述べる。第三章では関連発言を抽出するシステムを提案する。第四章では収集してアノテーションを付与したツイートに対して提案手法による実験及び考察を行う。第五章では全体に対する結論と今後の課題について述べる。

### 2. 関連手法

本研究で利用したデータベース及び関連手法について述べる。

## 2.1 日本語 Wordnet

日本語 Wordnet とは単語を Synset と呼ばれる類似概念情報でまとめ、保存したデータベースである。例えばデータベースには「弁護士」で検索を行うと「代言人」という類義語情報を提供する。また類義語情報だけではなく単語概念をより抽象的にした上位語、より具体的にした下位語などの情報も同梱されている。先ほどの弁護士を例に挙げれば上位語に“professional”があり、下位語に“divorce lawyer”などである。

## 2.2 TF-IDF

Bag-of-Words(BOW)はツイート用語順や構造を考慮せずに単語の頻度情報を利用してベクトル空間で表現する。しかし頻度情報だけだと、どのツイートにも高頻度で登場するような一般語なども重要性を上げてしまう。そこで Term Frequency - Inversed Document Frequency (TFIDF) は単語の特徴量を生成する際、2つの指標を用いて単語の特徴量を決定する。1つ目は単語 $w_i$ において一つのツイート $d$ 中にどれだけの頻度で登場するかを示す $TF_{i,d}$ 。また単語 $w_i$ が含まれるツイートが全体のツイートの中でどれだけあるかを示す $IDF_{i,D}$ 。 $IDF_{i,D}$ は全ツイート中で単語 $w_i$ が登場するツイート数を $f_{i,D}$ とし、全体のツイート集合を $D$ で表し、式2に示される。これら2つを式1で計算することで、他の文章にも頻出する単語の特徴量を下げ、その文章にのみ登場する単語の重要性を上げることが出来る。

$$w_{i,d} = TF_{i,d} * IDF_{i,D} \quad (1)$$

$$IDF_{i,D} = \log_2(|D|/f_{i,D}) \quad (2)$$

## 2.3 Latent Semantic Analysis

BOW, TFIDF を用いた場合、同一素性を含まない文章同士の類似比較が出来ず、間接的に関係する文章を見出すことが難しい。また共起情報を扱うことが出来ない。そこで Latent Semantic Analysis(LSA), もしくは Latent Semantic Index(LSI)は文章集合 $P$ と単語集合 $Q$ で作成した共起行列 $A$ に対し、特異値分解を行うことで前述の問題点を解決する。また松本ら[9]の解釈によると、視覚的に共起行列の関係性を把握する手助けをする。

まず式3の $A$ は行が全文章数 $p$ 、列が全単語数 $q$ で構成された共起行列である。 $r$ は $A$ のランクであり、 $\text{ランク}r \leq \min(p, q)$ である。 $U$ は $p \times p$ 、 $V$ は $q \times q$ の正方行列であり、 $D$ は特異値を対角要素に持つ $p \times q$ の対角行列、 $I$ は単位行列である。

$$A_{p \times q} = U_{p \times r} D_{r \times r} (V_{q \times r})^T \quad (3)$$

$$A = UDV^T, \quad UU^T = V^T V = I \quad (4)$$

ここで増加した計算量を減少させるため、元の性質を維持したまま $k$ 次元( $k \ll r$ )へ次元圧縮を行う。特異値を $\sigma_i$ で表すと、式6の近似の誤差を表すフロベニウスノルムが最小になるように $r-k$ 個の特異値を捨てることで式3に近似した式5のランク $k$ の $\tilde{A}_k$ が得られる。

$$\tilde{A}_k = U_k \tilde{D}_k V_k^T \quad (5)$$

$$\|A - \tilde{A}_k\|_F = \sqrt{\sum_{i=k+1}^r \sigma_i^2} \quad (6)$$

## 2.4 cosine 類似度

BOW, TFIDF によるベクトル量子化後の2つのツイート間における類似度をそれぞれの特徴ベクトル $\vec{v}, \vec{q}$ を用いて算出する。式7の値が1.0に近づくほど、類似性が高くなる。

$$\text{sim}(\vec{v}, \vec{q}) = \frac{\vec{v} \cdot \vec{q}}{\|\vec{v}\| \|\vec{q}\|} \quad (7)$$

## 3. 関連発言抽出手法の提案

大勢のユーザにリツイートされたツイートに関連する発言取得方法として、TwitterAPIを利用してツイートのJSONデータを取得する手段があるが、この仕様を利用しない人が多いため関連発言の取得精度が高くない。それは多くのユーザが通常のツイートと同様の方法で関連発言をツイートするためである。その結果、Twitterが用意したシステム上での関連発言の抽出はカバレッジが非常に低い。このような関連発言を抽出する手法を提案する。提案手法はツイート拡散の様子、関連ツイート収集部と関連推定アルゴリズム部に分けて解説する。

### 3.1 拡散の様子

ツイッターでは他のユーザを購読することをフォローと呼び、フォローした全ユーザのツイートは自身のHome\_timelineで閲覧することが可能である。図1において、 $User_A$ が投稿したツイートを拡散したユーザを灰色の丸で、拡散しなかったユーザを白色の丸で示す。 $User_A$ をフォローして購読している $User_B, User_C, User_D$ がいるとする。ある時、 $User_A$ が投稿したツイート内容を他者に伝えたいと感じた $User_B, User_D$ が情報拡散行為であるリツイートをする。その場合、 $User_A$ をフォローして購読していなくても、 $User_B$ をフォローして発言を購読している $User_E, User_F$ もしくは、 $User_C$ をフォローして発言を購読している $User_G, User_H, User_I$ は $User_A$ が投稿したツイートを読むことが可能である。また同様に $User_A$ の発言を $User_E, User_G$ が拡散して、 $User_E, User_G$ をフォローしているユーザへ伝達することが出来る。このように他者の発言を拡散することをリツイートと呼ぶ。Twitterではネズミ講のようにツイートがリツイートされ、情報がソーシャルネットワークに拡散されていく場合がある。この時、拡散の原因となった $User_A$ のツイートをシードツイートと以降では呼ぶ。また拡散したユーザはその時点でTwitterにログインしており、明らかに発言を購読したと考えられる。

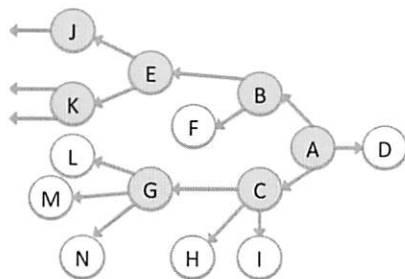


図1 各ユーザの拡散様子

### 3.2 拡散前後の関連発言数

ユーザの関連発言の投稿方法は主に二種類ある。1つ目はシードツイートを購読し、関連ツイートを投稿後、自分もリツイートする場合。2つ目はリツイートされた発言を購読し、自分もリツイート後、関連発言を投稿する場合である。

実際にどの程度、シードツイートをリツイートした後の前後で関連発言が存在するかを表1に示す。表1は2013年8月26日から2014年1月9日までにツイートを収集し、作成したテストコレクション7種である。4000A, 4000B, 4000Cはそれぞれ1000人のリツイート前後2件ずつのツイートに対して、シードツイートに関連しているかのアノテーションを付与したテストコレクションである。800A, 800B, 800C, 800Dは200人のリツイート前後2件ずつのツイートに対してアノテーションを付与したデータベースである。一般的にシードツイートをリツイートした1件後に関連ツイートを投稿するケースが多いが、800B及び800Dはこれに該当しない。それはシードツイートの内容が、現実社会で発生した問題に対して言及しているためである。このようなケースでは同事象に対する情報がTwitter上を飛び交い、ユーザが関連ツイートを投稿したり、関連ツイートをリツイートしたりするために継続的に関連ツイートが存在した。そのためリツイート2件前から2件後まで関連発言が増える結果となった。なお直接的にツイートに関係なくても、同事象について言及していたため、関係あるとしてカウントした。

表1 RT前後における関連発言数

名前	関連発言投稿タイミング			
	2件前	1件前	1件後	2件後
4000A	2	5	675	158
4000B	1	2	264	63
4000C	2	17	522	157
800A	6	12	49	15
800B	68	85	128	86
800C	0	0	25	7
800D	22	26	147	76

### 3.3 関連ツイート収集部

図2にシステムの概要図を示す。点線内で囲まれた部分の図中の数字2~6, は前述の拡散の様子と拡散前後の投稿発言を表す。関連ツイート取得部は取得システムとして描かれた図中の数字1, 7~9部分となる。

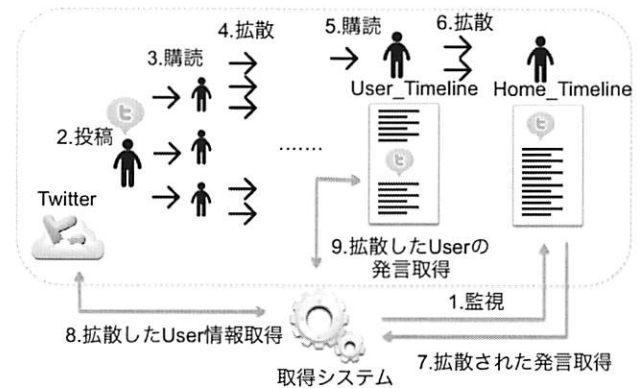


図2 ツイート取得システム図

取得システムのフローチャートを図3で示す。

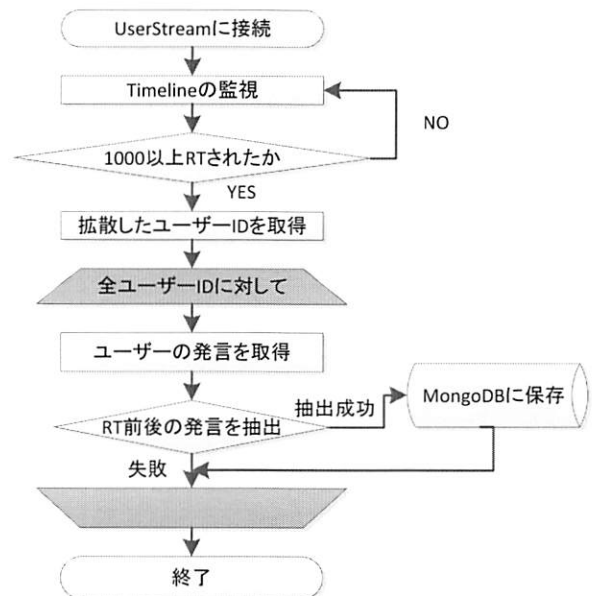


図3 取得システムフローチャート

本システム利用者の Home\_timeline を、リアルタイムで更新される TwitterAPI の UserStreams を利用して監視する。Home\_Timeline の監視中、1000人以上にリツイートされたツイートか検知を行う。リツイート数はツイートをJSON形式で取得しているため、キーの retweet\_count を参照することで判別する。判別されたツイートからリツイートしたユーザのID情報をTwitterAPIで複数回利用することで取得する。図1において灰色丸の拡散したユーザのみならず、User<sub>D</sub>などの白丸で囲まれたユーザもUser<sub>A</sub>のシードツイートを購読した可能である。しかし実際にこれらのユーザがTwitterにログインしてツイートを購入したかは不明であるため、これらのユーザの発言取得は行わない。

ユーザが投稿、もしくはリツイートしたツイートの一覧はそのユーザの User\_Timeline で購読が出来る。そこで各ユーザの User\_Timeline を最大800件取得参照し、もし最新のツイート3件目から798件以内にリツイートして

いた場合、リツイート前後 2 件ずつのツイートをリツイート内容に関連している可能性がある発言としてデータベースに保存する。

Twitter が提供する各種 TwitterAPI には一定時間内に実行可能回数の上限が設定されている。上限に達した場合は時間を置いて取得を行った。以上、上記の方法で取得したデータを解析部で利用する。

### 3.4 ツイート間の関連推定アルゴリズム部

2 ツイート間の関連推定アルゴリズムのフローチャートを図 4 に示す。

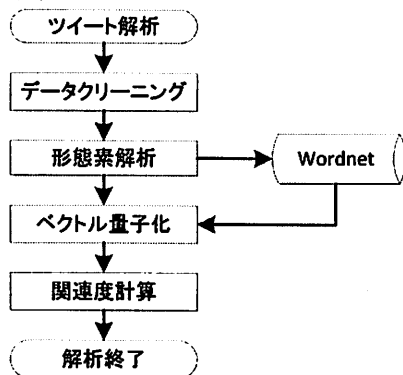


図 4 アルゴリズムフローチャート

取得部で取得した各ツイートに対して素性を作成するため、まず事前に形態素解析結果を悪化させる要因となる Twitter 特有の表現に、前処理であるデータクリーニングを行った。データクリーニングは特有な表現の例としてツイート本文に特定ユーザへの通知を行うためにツイート文頭に付与する @userID やリツイートを表す RT @userID:, ツイート中に記述された URL などの除去である。そのように前処理を行ったツイートに対して形態素解析器の Mecab を用いて形態素解析を行い、ツイートを構成する形態素を抽出する。本研究ではベクトル空間モデルを利用するため、素性のベクトル量子化には BOW, TFIDF, LSA の 3 手法のいずれかを用いる。2 ツイート間の関連推定計算には第 2.4 節で紹介したコサイン類似度を使用し、ツイートが拡散されたツイートに関係しているかそれぞれ算出した。またツイートは 140 文字以下で作成される素性が少ないという問題がある。素性不足を補うために、素性作成の段階で Wordnet を利用して素性の類義語を追加した。類義語の重複は除去を行った。

## 4. 実験

提案手法におけるシードツイートに関連するツイート抽出性能を各テストコレクションの実験にて測定する。実験はベクトル量子化 3 通り × Wordnet の有無 2 通り × 品詞 8 通り、計 48 通りの組み合わせで行った。

### 4.1 実験データ

2013 年 8 月 26 日から 2014 年 1 月 9 日までにツイートを収集し、7 つの別々のシードツイートとそれぞれに対して提案システムの収集方法を用いてツイートを収集し、

テストコレクションを 7 個作成した。収集したデータからリツイートした前後 2 件ずつのツイートに対して 4 種類のアノテーションを付与したテストコレクションを構築した。アノテーションは以下の 4 種類で、表 2 のとおりである。「1. 無し」は対象のシードツイートとの関係性は無い。「2. 有り」は対象のシードツイートに言及や想起された内容であり、関連している。「3. RT」は対象のシードツイートに関連しているが、他者のツイートをリツイートして関連している。「4. 曖昧」は関連性の有無の判別が困難で、曖昧に評価するならば関連している。節 3.2 で前述したように 4000A, B, C は 4000 件、800A, B, C, D は 800 件の計 15200 件にアノテーションを付与した。

表 2 アノテーションの内訳

名前	関係性の有無			
	1. 無し	2. 有り	3. RT	4. 曖昧
4000A	3160	810	9	21
4000B	3667	284	2	23
4000C	3302	594	82	22
800A	717	45	35	3
800B	433	175	180	12
800C	768	23	4	5
800D	529	136	120	15

### 4.2 アノテーション精度

実験データのアノテーション付与は一人で行われた。そのためアノテーション精度に対して疑問が残る。そこでテストコレクション 4000A の 300 件のツイートに対し、期間を一ヶ月以上設け、二度アノテーションの付与を行った結果を表 3 に示す。古いアノテーションと新しいアノテーションの一致度を測定し、精度を判断するために Kappa Statistic を用いた。式 8 より  $P(A)$  はアノテーション付与の一致率、 $P(E)$  は偶然にアノテーションが一致する確率を表す。その結果、 $k$  値 ( $kappa$ ) は 0.9879 であった。0.8-1.0 の値ならば good agreement とされるため精度の高いテストコレクションを構築したことが分かった。

表 3 Kappa Statistic

		旧アノテーション		
		関係有り	関係なし	Total
新	関係有り	51	1	52
	関係無し	1	248	249
	Total	52	249	300

$$kappa = \frac{P(A) - P(E)}{1 - P(E)} = 0.9879 \quad (8)$$

### 4.3 評価指標とベースライン

Precision は提案手法による関連性検索結果が表 2 における 2, 3, 4 を付与されたデータを正答しているかを示す。Recall は表 2 における 2, 3, 4 を付与されたデータを検索結果がどれだけ網羅しているかを示す。また

Precision, Recallの指標を総合的に解釈するための指標として F-measure(Fmeasure)と呼ばれる指標を式9に示す。

$$Fmeasure = \frac{2 * Precision * Recall}{Precision + Recall} \quad (9)$$

比較のベースラインとしてツイート TwitterAPI で取得した時の JSON 情報を利用した。ツイートにはそれぞれユニークな id が割り振られており、システム上で特定のツイートに対して文頭に相手の ID を含めた@userID を付与してリプライや非公式リツイートを送信する際に、多くのアプリケーションではユーザに秘匿して JSON キーの'in\_reply\_status\_id'及び'in\_reply\_status\_id\_str'の値に対象ツイートの id が与えられている。これらの情報、もしくははシステムのみに実装されていないアプリケーションでリプライを送付した可能性を考慮に入れ、本文中に相手ユーザの UserID がリプライ情報に含まれているかで、テストコレクション上で検索を行い、ベースラインの検索結果として扱った。表4にベースラインの F-measure とその時の Precision, Recall の値を示す。Precision が100%となると予想されたのにも関わらず、予想外に低かった原因として人間ではなく、プログラムが自動的に発言を行う Bot からのリプライがあったためである。また検索結果が0の場合は全ての値が0となっている。Recall が低いため、F-measure の値も Recall に引きずられて大きく低下した。

表4 既存手法の Fスコア

名前	ベースライン		
	Precision	Recall	F-measure
4000A	0.4	0.0023	0.0047
4000B	0.6666	0.0119	0.0233
4000C	1	0.0308	0.0597
800A	1	0.0609	0.1148
800B	0	0	0
800C	0	0	0
800D	1	0.0553	0.1048

#### 4.4 実験結果

第4.3節の指標とベースラインを用いて、提案手法の比較実験を行った。ベクトル量子化は BOW, TFIDF, LSI の3通り。Wordnet で類義語の追加有無で2通り。素性は NOUN(名詞のみ)、「VERB(動詞のみ)」「ADJ(形容詞の

み)」「N&V(名詞と動詞)」「V&A(動詞と形容詞)」「A&N(形容詞と名詞)」「N&V&A(名詞と形容詞と動詞)」「ALL(形態素解析器 Mecab によって生成される全品詞)」の8通り。これら3通り×2通り×8通り、計48通りの実験を各テストコレクションにおいて行った。

最も良い結果を示した手法、素性、その時のスコアを表5に示す。表4のベースラインと比較すると、提案手法は関連性のあるツイートのカバレッジが高いため Recall の値が良く、その結果 F-measure の飛躍的な向上を果たした。

ケーススタディとして図5、図6にテストコレクション 4000A の結果を示す。図5は横軸に検索数、縦軸に F-measure の値をとる。BOW+, TFIDF+, LSI+は Wordnet で類義語を加えた時の性能である。類義語を加えた場合は Wordnet のカバレッジが低く、またノイズとなりやすいために性能低下を招いた。

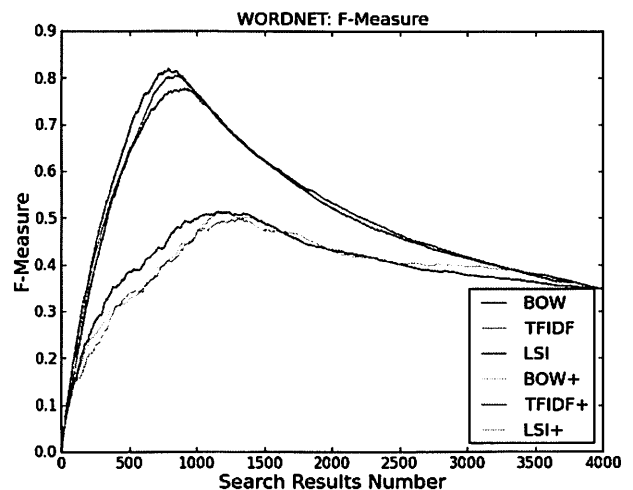


図5 4000A の名詞と形容詞素性の各手法性能

図6は各品詞素性における1件ずつ検索結果を増加させ4000件まで検索した際の Precision, Recall の関係を表す。横軸は Recall, 縦軸は Precision を示す。名詞素性を含む場合と含まない場合(動詞のみ, 形容詞のみ, 動詞と形容詞, 全品詞)でグラフがはっきりと二分している事がわかる。シードツイートの文字長は140文字丁度であった。

表5 各テストコレクションにおける提案手法の最大 Fスコア

名前	手法	素性	Precision	Recall	F-measure
4000A	BOW	形容詞+名詞	0.8455	0.7952	0.8196
4000B	TFIDF	名詞のみ	0.5409	0.4954	0.5172
4000C	TFIDF	名詞のみ	0.1931	0.7148	0.3041
800A	TFIDF	名詞のみ	0.8378	0.7560	0.7948
800B	BOW+	名詞のみ	0.8020	0.6403	0.7121
800C	TFIDF	形容詞+名詞	0.3124	0.2812	0.3
800D	TFIDF	形容詞+名詞	0.8119	0.6531	0.7239

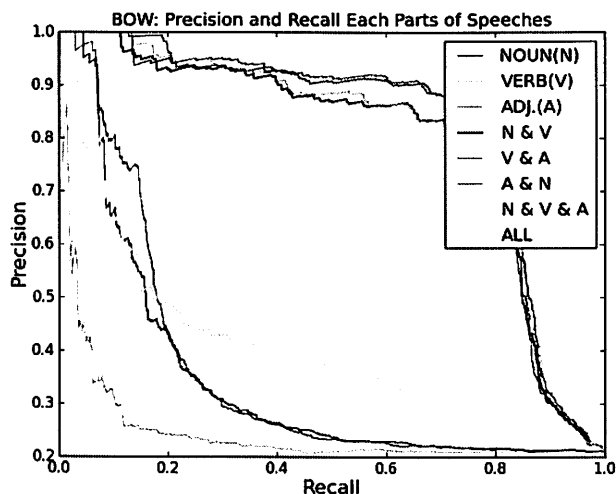


図6 4000AのBOWにおける各品詞素性の性能

#### 4.5 考察

各テストコレクションでの実験の結果、最適な手法は解析を行うテキストの性質や拡散したユーザの性質によって異なる。具体的や特徴的な固有名詞がシードツイートに含まれる場合は関連ツイートを投稿する際にその語を用いることあり性能が高い。逆にシードツイートの内容に対してユーザが詳しくない場合やシードツイートの素性が少なすぎた場合は性能が低くなった。

最も多く良い性能を発揮したのは名詞のみを素性としたTFIDFであった。その次に名詞と形容詞を素性としたTFIDFであり、名詞と形容詞を素性としたBOWと名詞とWordnetで名詞の類義語を素性として追加したBOWであった。名詞を基本とした場合は性能が高い。形容詞を追加した場合は関連ツイートの検索の網羅性が向上し、性能が向上する場合があった。しかし性能が向上はF-measureにおいて0.05未満であり、性能に悪影響をおよぼす場合のほうが影響は大きかった。また動詞は悪影響を及ぼしやすかった。LSIは期待されたようにシードツイートに含まれない素性でも関連性のツイートを検出してRecallの向上をもたらしたが、それ以上にPrecisionの性能低下を招き、F-measureで最も高い値を発揮することが出来なかった。また140文字以下というツイートの特徴においてMulti-gramはRecallの性能を大幅に下げただけになりやすい。

#### 5. まとめ

本稿ではTwitterにおける関連発言の抽出方法の提案と評価、データの分析を行った。提案システムはシードツイートをリツイートしたユーザのリツイート前後の発言を抽出し、関連性のあるツイートの抽出を行う。本システムの性能評価を行うためにシステムの取得部で取得したツイートを保存し、全15200件のツイートに対してアノテーションを付与して7種類のシードツイートのテストコレクションを作成した。その際実際にどの程度、前後に関連情報があるか調査したところ、最低でテストコ

レクションの4%、最大で45.8%であった。また、関連性のあるツイートと投稿までの時間間隔に強い相関性があり、シードツイートをリツイート後、関連のあるツイートを投稿する場合は、98%以上が1000秒以内に投稿されることが明らかとなった。この7種類のテストコレクションを用いて実験を行った。評価実験のベースラインにはツイートのJSONデータでシードツイートに関連しているかの情報を利用した検索結果を用いた。その結果、F-measureの最大値において提案手法が7種類全てのテストコレクションで既存システムの性能を大きく上回った。提案手法の中で最も多くのテストコレクションでいい結果を発揮したのは、Wordnetで類義語を追加せずに名詞のみを素性として利用したTFIDFであった。しかしシードツイートの内容やリツイートしたユーザによって最適な手法と素性が異なった。本システムの応用としてデマの拡散防止やマーケティングを考えている。今後は、他の様々なツイートでもテストコレクションを作成して実験を行い、ツイートと拡散したユーザの性質に応じて最適な手法を切り替える手法を開発することを考えている。

#### 参考文献

- [1] Twitter, <https://twitter.com/>, アクセス日(2014/2/2)
- [2] UNITED STATES SECURITIES AND EXCHANGE COMMISSION, <http://www.sec.gov/Archives/edgar/data/1418091/000119312513390321/d564001ds1.htm>, アクセス日(2014/2/3)
- [3] Thomson, R., Ito, N., Suda, H., Lin, F., Liu, Y., Hayasaka, R., ... Wang, Z. (2012). Trusting Tweets: The Fukushima Disaster and Information Source Credibility on Twitter, ISCRAM 2012, Vancouver Canada, pp.1-10.
- [4] Mendoza, M., Poblete, B., & Castillo, C. (2010). Twitter Under Crisis: Can we trust what we RT?, SOMA '10 Proceedings of the First Workshop on Social Media Analytics, pp. 71-79
- [5] 平成23年版 情報通信白書第一部, <http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h23/html/nc100000.html>, アクセス日(2014/2/3)
- [6] Doan, S., Vo, B. H., & Collier, N. (2011). An analysis of Twitter messages in the 2011 Tohoku Earthquake, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, 2012, vol. 91, Part 4, pp.58-66
- [7] Umejima, A., Miyabe, M. A. I., Aramaki, E., & Nadamoto, A. (2011). Tendency of Rumor and Correction Re-tweet on the Twitter during Disasters, 情報処理学会研究報告. データベース・システム研究会報告 2011-DBS-152 vol.4, pp.1-6.
- [8] Collier, N. and Doan, S. (2012), "Syndromic classification of Twitter messages", Lecture Notes of the Institute for Computer Science, Social Informatics and Telecommunications Engineering, vol. 91, pp. 186-195
- [9] 松本裕治(編)・相澤彰子・内山清子(2011), シリーズ朝倉〈言語の可能性〉6 言語と情報科学, 朝倉書店, pp. 67-74