

リツイートパターンに基づくツイートの到達確率判定と編集支援手法

松林 侑輝† 北山 大輔†

† 工学院大学情報学部コンピュータ科学科 〒163-8677 東京都新宿区西新宿 1-24-2

E-mail: tj113106@ns.kogakuin.ac.jp, kitayama@cc.kogakuin.ac.jp

あらまし Twitter を情報発信ツールとして活用している例は、個人や企業に関わらず多くなってきている。情報を拡散する手段としてリツイート機能(以下、RT)が存在するが、意図してRTをしてもらうことは難しく、思ったように情報が拡散できていない例は多く見られる。また炎上のように意図せず情報が拡散されてしまい、社会的な損失につながるようなケースも発生している。そこで本研究では、ユーザのRTパターンをナイーブベイズによって学習することで、任意のツイートのRT確率を推測し、よりRTされやすいツイートにするための編集支援手法を提案する。

キーワード Twitter, マイクロブログ, リツイート, 情報拡散, ナイーブベイズ

1. はじめに

近年、Twitter は情報発信ツールとして個人のみならず企業のPRとしても多く活用されている。NTT コムリサーチの調査 [1] によると Twitter を活用したいソーシャルメディアとしている企業の割合はいまだ増加傾向にあり、今後も情報発信のツールとして活用されると考えられる。

Twitter 上での情報発信は主にツイートという短文によって行われるが、特に RT という機能が Twitter 上における情報拡散の要となっている。これはユーザが自分のタイムラインに流れてきたツイートを、自分のフォロワーのタイムラインへと転送する機能である。この機能により自分をフォローしているフォロワー以外のユーザに情報を届けることが可能になっており、フォロワーを増やすきっかけでもあるため、情報を拡散したい・フォロワーの獲得をしたいユーザは RT されるような文章や話題をツイートすることが必須となる。しかし、意図的に RT されるようなツイートをするのは難しく、ツイートの受け手が RT したいと思わせる内容が伴っていないと難しい。

ユーザが RT するツイートに明確な基準は存在しないが、自分に関係する内容や、趣味嗜好に関連するものが一般的に RT されるツイートであると考えられる。また、文章の書き方によっても RT のされやすさが変化すると考えられる。例えばフォロワーの傾向やユーザ本人のキャラクター性から RT するツイートも固い表現のほうを好む場合や、逆にネットスラングを含むような砕けた表現を好む場合や、特定のキーワードが入っていると RT する場合などが考えられる。しかし、そうした特徴を何人にも渡って調査するのは困難である。

そこで本研究ではフォロワーの RT パターンを学習することで、RT 確率の推測を行い、より RT されやすい文章になるような編集支援手法を提案する。RT の確率の推測にはナイーブベイズを用い、ツイッターユーザとそのフォロワーのツイートを訓練データとした学習を行うことで、ツイッターユーザがツイートを RT するかどうかを判別する。編集支援手法ではツイートに類似語の置換・追加、投稿時刻・URL、メディアの有無・文字数の提案を行い、前述のナイーブベイズによる RT 分

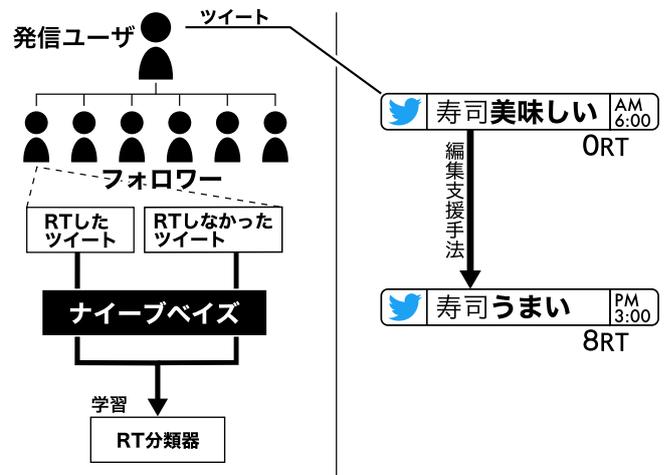


図1 概念図

類器で RT 確率が向上したか判別する。なお、類似語の獲得には Word2Vec を用いる。図1に本手法の概念図を示す。

本稿では2章で関連研究について述べ、3章ではナイーブベイズを用いた RT 確率の判別、4章ではツイートの編集支援手法、5章では実験とその考察について述べる。最後に6章でまとめと今後の課題について述べる。

2. 関連研究

近年、Twitter に関する研究は多数行われている。ツイートの分類の研究として、山本ら [2] は第一段階に教師なし学習である LDA を用いたトピック抽出、第二段階でラベル付けした少数のツイートをを用いることで、実生活ツイートのマルチラベル分類手法を提案した。また、[3] [4] のようにツイートの分類はこのようなトピックに着目したものが主流である。本研究では、過程としてツイートの分類を行うが、RT をするかしないかという分類を行う点でこれらとは異なる。また、RT に関する研究も多く行われている。山本ら [5] は ObjectRank を応用し、フォロワーをユーザの評価とするのに加え、RT をツイートへの評価とするユーザランキングの手法を提案した。川本ら [6] は多く RT されたツイートを SVM(サポートベクターマシン)

を用いることで、社会的影響力を持つツイートの早期検知を行う手法を提案している。これは訓練データに広範囲かつ大規模に取得したツイートを使用しており、対象ユーザとそのフォロワーからツイートを取得し、訓練データとしている点で本研究と異なる。中田ら [7] は、ツイートの文法的特徴の違いによって反響に違いが出るという仮説のもとに、SVM とロジスティック回帰によって影響力のあるツイートを推測する手法を提案した。これは文法的特徴によって影響力のある投稿の違いを推測することにとどまっておらず、どのような特徴が影響力に有効であるかを示すには至っていない。石垣ら [8] は、RT の回数と反応速度を考慮したネットワークによってユーザの重要度を推定する手法を提案した。これはツイートの内容を考慮せず、RT の回数と反応速度のみを使用している点で本研究とは異なる。

3. ナイーブベイズによるツイートの RT の判別

3.1 ナイーブベイズ

ナイーブベイズはベイズの定理に基づいた教師あり学習の分類手法で、ベイズの定理を応用した以下の式で求められる。

$$P(cat|doc) = \frac{P(cat)P(doc|cat)}{P(doc)} \quad (1)$$

$$\propto P(cat)P(doc|cat)$$

ナイーブベイズの学習には文書とそのカテゴリラベルを付与したものを訓練データとして用いる。 $P(cat)$ は事後確率でカテゴリ cat の文書数が訓練データの総文書数中で占める割合である。 $P(doc|cat)$ はカテゴリ cat が与えられたときに、文書 doc が生成される確率で、その確率は文書 doc 中の単語の出現確率の積によって求まる。文書 doc を与えたとき $P(cat)P(doc|cat)$ が最も大きくなるカテゴリがそのテキストのカテゴリとなる。本研究では、このナイーブベイズを用いたテキスト分類を用いて、ツイートの RT を判定した。次章で行う実験ではカテゴリラベルを「RT した」「RT しなかった」として学習を行っている。

3.2 訓練データ

ナイーブベイズの学習に用いるデータは、あるツイッターユーザが RT したツイートとそのツイッターユーザのタイムライン中の RT しなかったツイートを使用する。あるツイッターユーザが RT したツイートはユーザの最新ツイートから過去 n ツイートの中から抽出する。あるツイッターユーザが見ているタイムライン自体は Twitter API で取得することは出来ない。そこで、RT しなかったツイートは発信ユーザのフォロワーを取得し、各フォロワーの最新ツイートを m ツイートずつ取得し、擬似的なタイムラインとみなして、先に取得した RT したツイートに含まれなかったツイート群を RT しなかったツイートとする。

ツイートのテキストは形態素解析を行い、名詞・形容詞・形容動詞を抽出した。ツイートのメタデータ (投稿時間, RT, Fav, 文字数, URL の有無, メディアの有無) についてはその数値や文字列のまま訓練データとしてしまうと、その値自体との確率が計算されてしまう。そこでそれぞれユニークな語に置き換え、訓練データとした。例えば RT 数が 0 の場合は RT_NONE, RT

表 1 メタデータの扱い

メタデータ	ワード
RT	RT_NONE, RT_FEW, RT_MANY
Fav	FAV_NONE, FAV_FEW, FAV_MANY
時間帯	ZONE_MORNING, ZONE_NOON, ZONE_EVENING, ZONE_NIGHT, ZONE_MIDNIGHT
URL の有無	URL_EXISTS, URL_NONE
メディアの有無	MEDIA_EXISTS, MEDIA_NONE
文字数	SHORT_TWEET, LONG_TWEET

数が多い場合は RT_MANY というような語に設定した。詳しいメタデータの扱いについては表 1 に示す。

時間帯については 6~10 時台を朝, 11~14 時台を昼, 15~18 時台を夕方, 19~22 時台を夜, 23~翌 5 時台を深夜とした。なお、ユーザが RT したツイートについては Twitter API では RT した時間を取得することができないため、RT 直前のツイートと RT 直後のツイートの間の時間を RT したツイートの時間とし、RT したツイートが前後のツイートから 4 時間以上離れている場合は時間帯の特定ができないとして訓練データには追加しない。

また、文字数については訓練データ全体のツイート本文 (ユーザ ID, URL を除く) の文字長の中央値未満ならば「短いツイート」中央値以上ならば「長いツイート」とした。

以上の処理をしたものを訓練データとし、ナイーブベイズの学習を行った。

4. ツイートの編集支援手法

ツイートの編集支援として本研究では、ツイート本文の編集とメタデータの提案の 2 通りを行う。

4.1 ツイート本文の編集

ツイート本文の編集支援は置換・追加・削除などの編集操作に対して、その編集による RT 確率を表示することで行う。この時、置換や追加に関しては類似語・関連語を取得できれば、RT 確率を上げる語の推薦も可能となる。本稿では、類似語の獲得に Word2Vec を用いる。Word2Vec は Mikolov ら [9] によって提案された単語の特徴ベクトル獲得手法である。Word2Vec の学習にはナイーブベイズの学習に用いた訓練データ中のツイート本文を用いた。これはナイーブベイズの学習に用いた語以外が類似語として抽出されても、語の RT 確率がわからず、編集後のツイートの RT 確率が計算できないためである。ツイート本文は分かち書きを行い、動詞・形容詞・形容動詞については原型に戻した語を使用した。

提案する類似語の品詞は形容詞・形容動詞とした。これは名詞を類似語とした場合、文意が変わってしまうような語が類似語として抽出されることが多く、提案する語として適当でないと考えたためである。

4.2 メタデータの提案

ツイート本文の編集とは別に RT 確率の上昇するメタデータの提案を行う。提案するメタデータは、発信ユーザが制御できるメタデータである、時間帯・URL の有無・メディアの有無・

表2 ナイーブベイズ分類精度

追加したメタデータ	両ラベル	RTしたラベル	RTしていないラベル
全部	79.80%	75.43%	84.18%
ツイート本文のみ	59.31%	78.18%	40.44%
ユーザIDのみ	69.39%	84.81%	53.96%
RT数のみ	80.79%	86.35%	75.23%
Fav数のみ	68.69%	76.21%	61.16%
URLのみ	65.30%	67.97%	62.63%
メディアのみ	68.89%	74.90%	62.88%
時間帯のみ	67.18%	71.89%	62.46%
文字長のみ	67.22%	65.80%	68.65%
時間帯・URL・メディア・文字長	69.91%	61.65%	78.16%

文字数について行う。

5. 実験

5.1 ナイーブベイズによるRTの分類精度実験

テストデータを用いて、ナイーブベイズの分類精度の検証を行った。検証に用いるデータとして、あるユーザ10人分の最新ツイートからRTしたツイート、またそのユーザの疑似的なタイムラインからRTしていないツイートを抽出した。RTしたツイートからランダムに半分を抽出し「RTした」ラベルを付与する。また疑似タイムラインから「RTした」ラベルを付与したツイートと同数をランダムに抽出し「RTしなかった」ラベルを付与し、訓練データとした。テストデータには訓練データに使用していないRTしたツイートと、疑似タイムラインからテストデータにしたRTしたツイートを同数を抽出したものを使用した。なお、ツイート本文の形態素解析にはMeCab [10]を用いた。Twitterの特性上、ネットスラングや新語が多いため、辞書にはipadicを拡張したmecab-ipadic-NEologd [11]を利用した。

テストデータ計400ツイートに対して分類器でカテゴリラベルを判別し、正しく分類された割合を算出するのを1試行とし、それを20試行繰り返した。実験結果を表2に示す。

以下に考察を述べる。表2より「ツイート本文のみ」と「メタデータ全て」を訓練データとした場合を比較すると、「メタデータ全て」を訓練データとした場合のほうが分類精度が高くなっていることがわかる。また、素性ごとに着目すると「RT数」を訓練データとした場合にもっとも分類精度が高くなっていることがわかる。また、5.2章の実験でユーザが操作できるメタデータ「時間帯・URL・メディア・文字長」の結果に着目すると、「RTしたラベル」の分類精度は「ツイート本文のみ」の場合に劣るが、「RTしていないラベル」の分類精度は向上しており、全体として分類精度が高くなっていることがわかる。

5.2 ツイートの編集支援手法の実験

ツイートの編集支援手法の実験では「大事に使っていたので状態も良いです。」という文に対してRT確率の変化をみた。また、Twitterユーザの被験者にシステムを使用してツイートの編集を行ってもらい、システムの評価実験を行った。

表3 「大事」の類似語

語	コサイン類似度
大切	0.9945
非常	0.9931
ラフ	0.9929
綺麗	0.9927
ステキ	0.9923
無理	0.9922

表4 「良い」の類似語

語	コサイン類似度
上手い	0.9858
少ない	0.9809
高い	0.9808
にくい	0.9807
悲しい	0.9805
うまい	0.9804
美しい	0.9801

表5 「大事」を置換した場合のRT確率

語	RTする
大事	48.2233%
綺麗	49.1659%
大切	49.1659%

表6 修飾語「非常」を追加した場合のRT確率

語	RTする
追加なし	48.2233%
追加あり	48.5344%

5.2.1 ツイート本文編集の実験

本節ではWord2Vecを用いて、入力された語と同じ品詞の語を類似語とし、獲得した語を入力された語と置換、もしくは修飾語として追加を行う。その結果、RT確率がどのように変化するかをみた。

分類器の学習に用いたツイートはあるユーザの最新3,200ツイート中のRTしたツイート全て、およびあるユーザのフォロワー70ユーザから各200ツイートからユーザがRTしなかったツイートを使用した。Word2Vecの訓練データは上記のツイートをすべてを使用した。いずれの訓練データも5.1節と同様の形態素解析エンジンと辞書を用いて形態素解析、分ち書きを行った。

文中から形容動詞「大事」、形容詞「良い」の類似語として抽出された語、上位10件を表3、表4に示す。

表3よりそのまま置換できるような語として「綺麗」・「大切」という形容動詞の語が得られている。また、「大事」を修飾するような語として「非常」が得られている。

表4では文意に沿うような語が得られていないが、「上手い」・「うまい」・「美しい」のような肯定的な語が得られており、他の文の場合ならば置換できるような語となっている。次に、得られた「大事」の類似語「綺麗」・「大切」それぞれに置換した場合のRT確率を表5に示す。

この結果から「大事」を「綺麗」・「大切」に置き換えた場合の方が、RT確率が高くなっていることがわかる。次に「非常」を「大事」の修飾語として追加した場合について表6に示す。

結果より、「非常」を修飾語として追加した場合の方がRT確率が高くなっていることがわかる。

5.2.2 メタデータ提案の実験

この節ではメタデータごとのRT確率について述べる。実験を行ったメタデータは時間帯・URLの有無・メディアの有無・文字長である。時間帯によるRT確率の変化を表7に示す。結果より、深夜帯にもっともRT確率が高くなることがわかった。

次に、URLの有無によるRT確率を表8に示す。結果より、URLを含むツイートのほうがRT確率が高くなることがわ

表7 時間帯による RT 確率

時間帯	RT する確率
朝	48.3338%
昼	48.1270%
夕方	48.2302%
夜	48.2304%
深夜	49.1590%

表8 URLの有無による RT 確率

URLの有無	RT する確率
URL なし	48.3776%
URL あり	48.5084%

表9 メディアの有無による RT 確率

メディアの有無	RT する確率
メディアなし	48.1437%
メディアあり	49.0655%

表10 文字長による RT 確率

文字長	RT する確率
SHORT_TWEET	50.3468%
LONG_TWEET	48.9848%

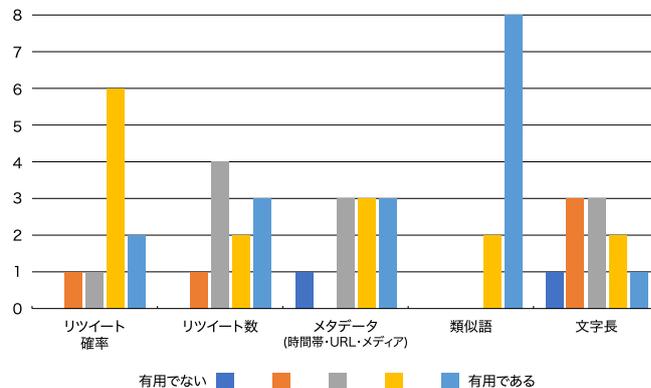


図3 実験結果

段階で行った、それぞれの項目について説明する。RT 数の表示は、文章を複数のユーザの分類器で判別し、「RT する」と判別された数を表示したものである。RT 確率の表示は、複数のユーザの分類器中でもっとも高い RT 確率を表示したものである。メタデータの提案は、もっとも RT されやすい時刻やメディア・URL の有無について提案を行う。類似語の提案では 5.2.1 節の類似語の提案を、入力された文章中の語に対して行った。文字長の提案では、RT されやすい文章の長さを提案した。

このシステムを用いて Twitter ユーザの被験者 10 人に対して評価実験を行った。実験結果を図 3 に示す。

5.2.4 考察

Word2Vec を用いて置換する類似語を提示する手法については候補として利用できる語が少数だが得られた。抽出した類似語の例として 2 例 (表 3, 表 4) あげたが、適当な類似語が得られていても候補数が少なく、文意に沿う語が得られるとは限らない。そのため、コーパスのデータ量を増やすことで抽出される語の数を増やす必要がある。今回、RT したツイートの抽出元のツイート数は Twitter API の制約上 3,200 ツイート、RT しなかったツイートの抽出元のツイート数がフォロワー数 × 200 ツイートであった。Search API を使用して 3200 件以降の取得・フォロワーごとの取得ツイート数を増やすなどを行うことで獲得できる類似語候補も増やすことができるだろう。

実験では、メタデータごとの RT 確率の変化も確認した。各結果から「深夜帯」に「URL」と「メディア」ありの「短いツイート」がもっとも RT 確率が高くなるメタデータの組み合わせであることがわかった。Word2Vec で獲得した類似語に置換した文に対して上記のメタデータを付与したところ、RT 確率が 50% を超え、「RT する」に判別されるツイート文が得られた。このように RT 確率に基づいて、メタデータの提案およびツイートの編集が可能であることを確認した。

時間帯についてはユーザが主にツイートしている時間帯を調査したところ、23 時と 24 時台にもっともツイートをしていた。よって今回対象としたユーザではもっともツイートしている時間帯、すなわちもっともタイムラインを閲覧している時間帯に RT がなされていた。今後はツイートがなされている時間とよく RT をする時間に違いがあるユーザがいるか調査することで、RT した時間を訓練データとするほうが有意であるか確認する

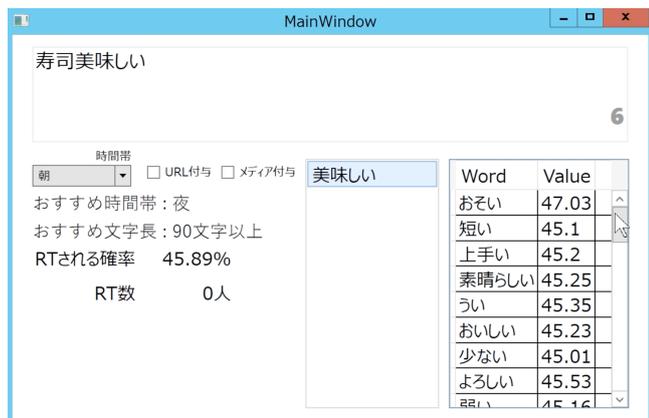


図2 システム図

かった。

次に、メディアの有無による RT 確率を表 9 に示す。結果より、メディアを含むツイートのほうが RT 確率が高くなることわかった。

次に、文字長による RT 確率を表 10 に示す。文字長についてはユニークな語として追加した語「SHORT_TWEET」「LONG_TWEET」の確率を示した。結果より短いツイートの方が RT 率が高くなることわかった。

以上の結果より、「深夜帯」かつ「URL あり」かつ「メディアあり」かつ「短いツイート」のツイートがもっとも RT 確率が高くなるメタデータの組み合わせであることがわかった。ここで 5.2.1 節の結果から得られるツイート本文「大切に使用したので状態も良いです。」という文に対して、上記のメタデータを追加したところ、RT 確率は 50.2971% となり、「RT する」に判別された。

5.2.3 システムの評価実験

本研究では、図 2 のようなシステムを作成した。ユーザはテキストボックスにツイートする文章を入力する。それに応じて RT 確率や RT 数の値の変化するため、ユーザはそれらを参考にしてツイートの編集を行うことができる。評価は「RT 数の表示」・「RT 確率の表示」・「メタデータ (投稿時刻・メディア・URL) の提案」・「類似語の提案」・「文字長の提案」に対して 5

文 献

必要がある。

システムの評価実験では、各要素に対してユーザに評価してもらった。図3より、RT 確率・RT 数の表示、類似語・メタデータの提案については有用であるという評価が得られた。一方で、文字長の提案についてはあまり有用でないという評価となった。これについては本システムでの文字長の提案は「訓練データ中の文字長の平均値以上か以下か」という提案であったため、ユーザの感覚にあった提案でなかったことが原因であると考えられる。

6. まとめと今後の課題

本研究では、ナイーブベイズを用いた RT 判別とツイートの編集支援を提案した。

ナイーブベイズを用いた RT 判別では、ユーザとそのフォロワーのツイートを訓練データとした学習を行った。また、ツイートが持つメタデータを訓練データとして加え、メタデータが分類精度にどの程度影響を与えるかを実際のツイートを用いた実験を行った。実験の結果、メタデータを学習に加えることで RT の分類精度が向上することがわかった。

ツイートの編集支援ではツイート本文の編集とメタデータの提案を行った。ツイート本文の編集支援は編集操作に対して RT 確率を表示することでを行い、語の置換・追加については類似語・関連語を Word2Vec を用いることで獲得する手法を提案した。また、メタデータの提案では投稿時間・文字長・URL・メディアの有無といった、ツイート本文以外の RT 確率を上昇させる要素について提案した。これらについても実験を行い、RT 確率に基づいてツイート本文の編集とメタデータの提案が可能であることを確認した。また、それらを用いたシステムを作成し、Twitter ユーザを被験者としたシステムの評価実験を行った。その結果、RT 確率・RT 数の表示、類似語・メタデータの提案の有用性を確認した。

今後の課題としては、トピックを考慮した RT 分類器の生成があげられる。トピックごとの傾向として、例えば食べ物に関するツイートであれば画像があるほうが RT されやすい、政治に関するツイートであれば文章が長いほうが RT されやすいというような傾向があると考えられる。本研究では、訓練データ全体での傾向が訓練結果となっているため、トピックを考慮することでよりユーザの感覚にあった RT 分類が可能になると考えられる。またフォロワーのフォロワーのように探索する幅を広げる場合、取得するツイート数が爆発的に増えることが考えられる。そうした場合、Twitter API の制約上すべてユーザのツイートを取得することが現実的に難しい。そこで発信ユーザのフォロワーの RT 確率でフォロワーのフォロワーの場合の RT 確率を近似するといった手法が考えられる。

謝 辞

本研究の一部は、平成 28 年度科研費若手研究 (B)(課題番号: 15K16091) によるものです。ここに記して謝意を表すものとします。

- [1] NTT コムリサーチ. 第 7 回企業におけるソーシャルメディア活用に関する調査. <http://research.nttcoms.com/database/data/001978/>.
- [2] 山本修平, 佐藤哲司. 二段階抽出法を用いた実生活 tweet のマルチラベル分類. Vol. 2013, pp. 64–71, 2013.
- [3] 京介西田, 遼平坂野, 考藤村. データ圧縮によるツイート話題分類. 日本データベース学会論文誌, Vol. 10, No. 1, pp. 1–6, 2011.
- [4] 塚田文哉, 鈴木徹也. Twitter におけるツイート間の反応を考慮した話題分類手法. Vol. 2014, No. 1, pp. 109–110, 2014.
- [5] 祐人山口, 俊之天笠, 翼高橋, 博之北川. 情報伝搬を考慮したグラフ分析による twitter ユーザランキング手法. 情報処理学会論文誌データベース (TOD), Vol. 4, No. 2, pp. 142–157, 2011.
- [6] 貴史川本, 正史豊田, 直樹吉永. マイクロブログにおける社会的影響力を持つ情報カスケードの早期検知に向けて. Vol. 2015, pp. 48–55, 2015.
- [7] 侑輝中田, 英史上岡. Twitter における影響力の分析手法. 研究報告ユビキタスコンピューティングシステム (UBI), Vol. 2014, No. 37, pp. 1–6, 2014.
- [8] 石垣藍睦, 沼尾雅之. Twitter 特有のネットワーク構造を用いたユーザ重要度評価法の提案. *DEIM Forum 2016 B7-4*, pp. 1–8, 2016.
- [9] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pp. 3111–3119. 2013.
- [10] Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. Applying conditional random fields to japanese morphological analysis. pp. 230–237, 2004.
- [11] Sato Toshinori. Neologism dictionary based on the language resources on the web for mecab, 2015, <https://github.com/neologd/mecab-ipadic-neologd>.