TwitterのIndirect Tweetを起点とした興味情報収集手法の提案

山本　直史† 　小林　亜樹††

† 工学院大学大学院工学研究科電気・電子工学専攻　〒163–8677 東京都新宿区西新宿 1-24-2
†† 工学院大学情報学部情報通信工学科　〒163–8677 東京都新宿区西新宿 1-24-2
E-mail: †cm15029@ns.kogakuin.ac.jp, ††aki@cc.kogakuin.ac.jp

あらまし　Twitterには，Indirect Tweetと呼ばれる投稿がある．Indirect Tweetは，暗黙的な繋がりを他のユーザあるいは投稿に持つ．そのため，ユーザはTL上に現れるIndirect Tweetの情報に興味を持ったとしても，対象とする投稿の収集や全体の話題を把握することは難しい．本研究では，文字情報と時間情報によってIndirect Tweetから言及先投稿の収集を試みる．文字情報ではユーザ間の関連度および投稿間の関連度を計算する．時間情報では過去のIndirect Tweetと言及先投稿との時間差によって関連度を計算する．実験では，あるユーザを中心とした相互フォロー関係にあるコミュニティの投稿を対象とした．

キーワード　SNS，Twitter，エアリプ，Indirect Tweet

1. はじめに

Twitterでは日々様々なコミュニケーションがなされている(注1)．Twitterは，各ユーザが手軽に情報発信者になれるという特徴がある．そのため，他ユーザのTweetを見たユーザがそのTweetに対して新しくTweetを行うことで，ユーザ同士のコミュニケーションも盛んに行われている．

Twitter社から提供されているコミュニケーションの機能としては，ユーザまたはTweetに対して言及をする機能としてMentionおよびReplyが存在し，Tweetの再拡散を行う機能としてRetweetやQuote Tweetが存在する．

一方，ユーザの慣習によって生まれたIndirect Tweet(エアリプ)というコミュニケーションの手段がある．Indirect Tweetはユーザ間でコンテキストを共有していることを前提とするようなコミュニケーションの一種であり，いわばある特定のTweetやユーザに対するMentionあるいはReplyでありながら，関連するユーザやTweetが暗黙的なTweetを指すものである．

Twitterにおいて，コミュニケーションによって発生する情報や他のユーザによる自分と同じあるいは異なる意見を持つTweetを収集する活動は支持されている．しかし，先程述べたIndirect Tweetについては，他の機能と同じくコミュニケーションによって発生する情報源でありながら，関連するTweetを収集しようとするサービスや先行研究はほとんど存在しない．

また，Indirect Tweetにはユーザの観点からも問題がある．Indirect Tweetを用いたコミュニケーション手段では，しばしばやりとりされるTweetの一部だけがユーザに見えてしまうのである．Indirect Tweetの特徴として，2人以上のユーザのコミュニケーションにおいて用いられるものでありながら，その内のいずれか1人をフォローしているだけで他ユーザのHomeTimelineに現れてしまう点が挙げられる．

(注1)：日本において Reply および Retweet は Tweet 全体の 50%ほどを占める．http://teapipin.blog10.fc2.com/blog-entry-680.html

Indirect Tweetを見たユーザは，Tweet内容に興味を持ち関連するTweetを収集したいと考えたとしても，興味を持ったTweetがどのユーザに対して行われているのか，どのTweetと関連があるのか把握することは出来ない．ユーザが関連するTweetを収集するには，興味を持つTweetをしたユーザとフォロー関係にあるユーザのTweetすべてを調べる必要がある．

本研究ではこのようなIndirect Tweetに関連するTweetを収集する難しさを解決するために，興味を持ったIndirect Tweetより以前にTweetされた，関連する可能性のあるTweet群に対して順位付けを行い，ユーザの興味情報の収集を効率化することを考える．本稿ではその実験として，ある特定のユーザを中心とした相互フォローによる集団のTweetを一定期間収集し，その中から人がIndirect Tweetと判断したTweetと，Indirect Tweetに関連する可能性のあるTweet群との関連度を計算し，順位付けを行う課題を設定する．

2. 問題設定

ここでは，Twitter上で投稿されるIndirect Tweetと言及先TweetおよびそれにかかわるユーザのFollow関係を定式化し，対象とする問題とその解決方法を明確にする．

2.1 Indirect Tweet

Simple Tweetの一種．Simple Tweetとは，他のユーザやTweetが言及先として存在しないあるいは明示的でないTweetのことを指す．Indirect TweetとはReplyやMentionの様に，他のユーザやTweetを言及先としているが，その言及先が明示的でないTweetを指す．言及先は，Indirect Tweetを投稿したユーザ以外にはTweetの内容を見て推測するしかない．

Indirect Tweetは他のTweetを見た際に投稿される．Indirect Tweetが投稿されるような状況は幾つか考えられる．これらは，Indirect Tweetの投稿されやすさやIndirect Tweetを投稿したユーザとフォロー関係があるか，Tweetの種類がなにであるかという違いがある．本稿では，HomeTimeline上に表示される相互フォローのユーザのTweet，Reply，Mention

を言及先とする場合の Indirect Tweet のみを取り扱う．また，Indirect Tweet は事前に人手で定めたもののみを取り扱うが，その指定が妥当であるかという点については議論しない．

2.2 言及先 Tweet

本稿では，Simple Tweet，Reply，Mention のうち，前述の Indirect Tweet の言及先である Tweet のことを言及先 Tweet と呼ぶ．ある Indirect Tweet の言及先 Tweet は 1 つであるとは限らない．例えば，あるユーザの連続的な Tweet に対して Indirect Tweet が投稿されたのであれば，その連続的な Tweet それぞれが言及先 Tweet だと考える．また，2 人以上のユーザ間での Reply や Indirect Tweet に対して Indirect Tweet が投稿されればそれら複数の Tweet が言及先 Tweet となる．

2.3 文字定義

サービスの利用者 u

SNS のユーザを記号 u で表し，その全体集合を U とする．

Tweet $p = (\text{user}, \text{time}, \text{text}, \text{class}, \text{number})$

SNS における，ある Tweet を p で表す．ただし，パラメータ user は Tweet p_i を投稿したユーザの ID を表し，パラメータ time は Tweet p_i の投稿時刻を表し，パラメータ text は Tweet p_i の本文を表す．パラメータ class は Tweet の分類を表す．type には，simpletweet，reply，mention，retweet，quotetweet が存在する．パラメータ number は user にとって何番目の Tweet であるかを表すものとして，最新の Tweet から順に $1, 2, ...$ という値を取る．Tweet の全体集合を $P = \{p_1, p_2, ...\}$ とする．

フォロイー集合 F_u

あるユーザ u について，そのフォロイー集合を F_u と表す．ユーザ ID が i であるユーザの HomeTimeline を表す Tweet 集合 P_i とするとき，式 (1) で表される．

$$P_i = \{p_j \mid \mathrm{Id}(p_j) = i \cup \mathrm{Id}(p_j) \in F_i\} \tag{1}$$

$\mathrm{Id}(p)$ は任意の Tweet からユーザ ID を求める関数であり，与えられた Tweet から user を返すものとして，式 (2) で表される．

$$\mathrm{Id}(p) = \text{user} \tag{2}$$

関数 $\mathrm{Type}(p)$

任意の Tweet からその分類を求めるための関数であり，Tweet から class を返すものとして，式 (3) で表される．

$$\mathrm{Type}(p) = \text{class} \tag{3}$$

関数 $\mathrm{Time}(p)$

引数として任意の Tweet を与えると，その投稿時刻を返す関数．ただし，この投稿時刻は基準となる現在時刻よりどれだけ時間軸上において離れているかを表したものである．すなわち，現在時刻より離れているほど大きくなるものである．

$$\mathrm{Time}(p) = \text{time} \tag{4}$$

関数 $\mathrm{List}(P, p, n)$

Tweet によって構成される集合から指定された Tweet を取り出す関数．取り出される Tweet は，集合の元において p よりも時間軸上で手前に存在する最も新しい Tweet から順に n 番目の Tweet になる．

Indirect Tweet および言及先 Tweet

Indirect Tweet はあるユーザがフォローされている時に，そのフォロワーによって Tweet されるものである．すなわち，あるユーザをそれぞれ u, v として $v \in F_u$ であるとき，Tweet $p_i \in P$ が $\mathrm{Id}(p_i) = u \wedge \mathrm{Type}(p_i) = \text{simpletweet}$ である場合，Indirect Tweet である可能性がある．また，Indirect Tweet は 1 つのみが存在するわけではない．ここで，任意の Tweet 集合に属する Indirect Tweet すべてを $\mathrm{I}(P)$ なる関数によって求められるものとする．このとき，$p_i \in P$ なる Tweet について $p_i \in \mathrm{I}(P)$ である場合を p_i は Indirect Tweet であるとする．

ここで，任意の Indirect Tweet p_i には必ずその言及先 Tweet が存在する．$v \in F_u$ であり，$p_i \in \mathrm{I}(P)$ であるとき，P に属する $\mathrm{Id}(p_j) = v \wedge \mathrm{Type}(p_j) = (\text{simpletweet} \vee \text{reply} \vee \text{mention})$ なる Tweet は言及先 Tweet である可能性がある．任意の Tweet 集合に属し，p_i なる Indirect Tweet のすべての言及先 Tweet を $\mathrm{R}(p_i, P)$ なる関数によって求められるものとする．すなわち，$p_i, p_j \in P$ として，$p_i \in \mathrm{I}(P)$ が成り立つのであれば，$\exists p_j \in \mathrm{R}(p_i, P)[\mathrm{Id}(p_j) = v \wedge \mathrm{Type}(p_j) = (\text{simpletweet} \vee \text{reply} \vee \text{mention})]$ である．また，Indirect Tweet の言及先 Tweet が 1 つのみであるとは限らない ($|\mathrm{R}(p_i, P)| \geqq 1$)．

$p_i \in \mathrm{I}(P)$ で，$p_j \in \mathrm{R}(p_i, P)$ であるとき，時間軸において常に $\mathrm{Time}(p_i) < \mathrm{Time}(p_j)$ が成り立つ．すなわち，

$$p_j = \mathrm{List}(P, p_i, j - i) \tag{5}$$

である．

2.4 Indirect Tweet の閲覧における問題点

ユーザ $u, v, \nu \in U (u \neq v, u \neq \nu, v \neq \nu)$ について $u \in F_\nu$ かつ，$v \notin F_\nu$ であるとする．また，$u \in F_v$ かつ $v \in F_u$ である．いま，$p_i \in P, \mathrm{Id}(p_i) = u$ について $p_i \in \mathrm{I}(P)$ であるとすると，ユーザ ν の HomeTimeline P_ν は $p_i \in P_\nu$ であるので，Indirect Tweet を閲覧することになる．ここで，$p_j \in P, \mathrm{Id}(p_j) = v$ について $p_j \in \mathrm{R}(p_i, P)$ だとしたとき，ユーザ ν の HomeTimeline P_ν において $p_j \notin P_\nu$ であるので，Indirect Tweet である p_i の言及先 Tweet p_j は閲覧されない．

ユーザ ν は Indirect Tweet p_i に言及先 Tweet p_j が存在することが予想できたとしても，言及先 Tweet を投稿したユーザ v を F_u より探し出すのは難しい (注2)．そのため，ユーザ ν は日常的に閲覧する HomeTimeline P_ν において，言及先 Tweet が明示的でない不完全な情報が与えられることになる．

この問題はユーザ自身で解決するのは難しく，またユーザにとっては日常的に閲覧している HomeTimeline で偶発的に発生しうる問題である．そのため，本研究では Indirect Tweet である p_i の言及先 Tweet p_j を P_u より効率的に収集するための手法を提案する．言及先 Tweet p_j を投稿したユーザ v は F_u に属する．そのため，F_u に属するようなユーザの Tweet について，p_i との関連度を計算することになる．本稿では特に，u

(注2)：ライフメディアのリサーチバンクによると 2015 年時点で 10 人以上のユーザをフォローしているユーザが 80%程.
http://research.lifemedia.jp/2015/05/150513_twitter.html

と相互フォローの関係にあるユーザのみを対象として，Simple Tweet，Reply，Mention との関連度を計算する．

3. 提案手法

本章では，Indirect Tweet の言及先 Tweet を効率的に収集する目的のために，任意の処理対象 Tweet に対して，時間軸上で前の時間に存在する計算対象 Tweet 群に対して関連度によって順位付けを行う手法を提案する．本手法の概要図を図 1 に示す．本手法では，文字情報および時系列情報を用いて 3 つの関連度を求め，その総和によって順位付けを行うものである．

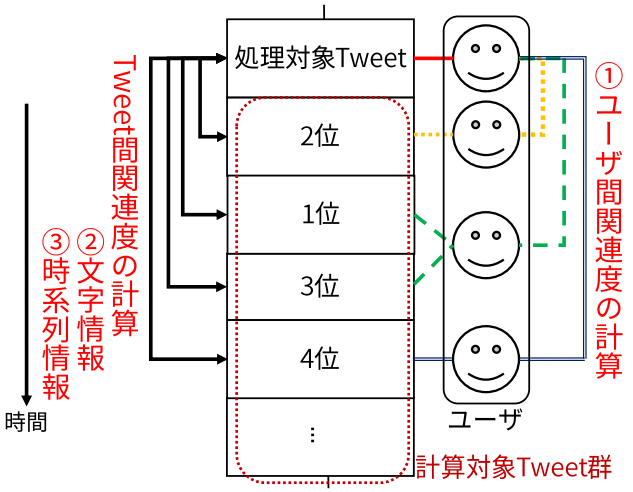

図 1　提案手法概要図

3.1 文字定義

フォロワー集合 E_u

ユーザ $u, v \in U$ とするとき,$v \in E_u$ であれば，v は u のフォロワーであり，u は v のフォロイーである．

相互フォロー集合 M_u

ユーザ u とフォロイーかつフォロワーであるユーザが属する集合．フォロー集合 F_u とフォロワー集合 E_u の積集合となる．

関数 Text(p)

任意の Tweet からその本文を求めるための関数であり，与えられた Tweet から text を返すものとして，式 (6) で表される．

$$\mathrm{Text}(p) = \mathrm{text} \tag{6}$$

Tweet 集合 T_u

あるユーザ u の Tweet p が属するような集合を Tweet 集合 $T_u = \{p_i \mid \mathrm{Id}(p_i) = u\}$ とする．UserTimeline を構成する Tweet 群と同義である．

関数 Number(u, p)

Number(u, p) はあるユーザ u について，任意の Tweet p が最新から何番目の投稿であるかを求める関数であり，与えられた Tweet から number を返すものとして，式 (7) で表される．

$$\mathrm{Number}(u, p) = \mathrm{number} \tag{7}$$

名詞集合 N_u

単一の Tweet，あるいは複数の Tweet の名詞を元とする集合．あるユーザ u の複数の Tweet からなる場合，N_u で表す．

関数 Noun(p)

Tweet p を与えると，p の名詞を要素とする集合を返す関数．

$$\mathrm{Noun}(p) = N \tag{8}$$

関数 Jaccard(N_a, N_b)

単語を要素とする 2 集合間の Jaccard 係数を求める関数．ある 2 つの名詞集合を N_a, N_b とする時，引数にこれらの名詞を要素とする集合を与えた場合を考えると，式 (9) のようになる．これは，2 つの集合の大きさに対する共通要素数の割合である．

$$\mathrm{Jaccard}(N_a, N_b) = \frac{|N_a \cap N_b|}{|N_a \cup N_b|} \tag{9}$$

関数 Simpson(N_a, N_b)

単語を要素とする 2 集合間の Simpson 係数を求める関数．式 (10) で表される．これは，要素数の小さな集合の大きさに対する共通要素数の割合である．ただし，$|N_a|$ または $|N_b|$ が 0 であるとき，この関数は 0 の値を返すものとする．

$$\mathrm{Simpson}(N_a, N_b) = \frac{|N_a \cap N_b|}{\min(|N_a|, |N_b|)} \tag{10}$$

関数 TimeSim(c_a, c_b)

引数として 2 つの時刻を与えると，単位時間における差分時間を基にした関連度を返す関数．c_a, c_b はある特定の時刻を指すものとし，差分は $|c_a - c_b|$ によって求められるものとする．

3.2 ユーザ Tweet によるユーザ間関連度の計算

ここではユーザ間の趣味嗜好の違いを反映するためのユーザ間関連度を算出する．本手法では，処理対象 Tweet を投稿したユーザとそのユーザと相互フォローの関係であるユーザについて処理を行う．即ち，処理対象 Tweet を投稿したユーザ $u \in U$ を考えた時，相互フォロー集合 M_u に属するユーザを関連度計算の対象ユーザとする．あるユーザ $v \in M_u$ として，v の Tweet 集合は式 (11) のようになる．

$$T_v = \{p_i \mid \mathrm{Id}(p_i) = v, v \in M_u\} \tag{11}$$

Tweet 集合 T_v において，一定の件数の Tweet について，その名詞を求める．すなわち，ユーザ v の名詞集合

$$N_v = \{\mathrm{Noun}(p_i) \mid p_i \in T_v, \mathrm{Number}(v, p_i) \leqq k\} \tag{12}$$

である．ただし，k は任意の自然数である．同様に，処理対象 Tweet を投稿したユーザ u についても式 (11),(12) の名詞集合を求める処理を行い，u から得られた名詞集合 N_u とする．

ここで，処理対象 Tweet を投稿したユーザ u と相互フォロー集合 M_u に属するユーザとで，名詞集合からユーザ間関連度を計算する．任意のユーザ $v \in M_u$ として，

$$j_v = \mathrm{Jaccard}(N_u, N_v) \tag{13}$$

となる．ここで，j_v はユーザ u と v のユーザ間関連度を表す．

3.3 Tweet 内容による内容的 Tweet 間関連度の計算

ここでは，Indirect Tweet と言及先 Tweet の間で同じ単語を用いる可能性が高いことに着目した関連度の求め方について

説明する．$p_i \in P$ なる処理対象 Tweet があるとするならば，その計算対象 Tweet 群は

$$\{p_j \mid p_j = \mathrm{List}(P, p_i, l), \mathrm{Id}(p_j) \in M_u\} \tag{14}$$

で表される．ただし，$l = 1, 2, ..., n(n$ は任意の自然数) である．処理対象 Tweet と計算対象 Tweet 群の間で内容的 Tweet 間関連度を算出するために，名詞集合を求める．すなわち，

$$N_i = \mathrm{Noun}(p_i) \tag{15}$$

$$N_j = \mathrm{Noun}(p_j) \tag{16}$$

である $(i < j)$．処理対象 Tweet p_i の名詞集合である N_i と計算対象 Tweet 群に属する任意の Tweet p_j の名詞集合である N_j とで Simpson 係数によって Tweet 間関連度

$$s_j = \mathrm{Simpson}(N_i, N_j) \tag{17}$$

を求める．s_j は p_i と p_j の内容的 Tweet 間関連度を表す．

3.4 投稿時刻による時間的 Tweet 間関連度の計算

ここでは，Indirect Tweet と言及先 Tweet の時間差にはユーザ毎に偏りがあることに着目した関連度について説明する．3.3 節と同様に処理対象 Tweet と計算対象 Tweet 群が定められるとき，p_i および $\{p_j\}$ に属する各 Tweet の投稿時刻を求め，

$$c_i = \mathrm{Time}(p_i) \tag{18}$$

$$c_j = \mathrm{Time}(p_j) \tag{19}$$

となる．関数 TimeSim() を用いて各 Tweet の関連度は

$$t_j = \mathrm{TimeSim}(c_i, c_j) \tag{20}$$

となる．ここで，t_j は処理対象 Tweet である p_i と計算対象 Tweet 群に属する任意の Tweet p_j との時間的 Tweet 間関連度を表す．Timesim() の計算過程は次節にて説明する．

3.5 処理対象 Tweet からの時間的距離を用いた関連度

筆者らは Reply が時間的に離れた Tweet に対して頻繁には行われないことから，Indirect Tweet についても同様の性質が存在すると仮定した．あるユーザについて収集した過去の Indirect Tweet と言及先 Tweet の時間差を正解データとして用いることで，時間的距離によって変化する関連度を設定する．本手法では，単位時間ごとに対してベイズ推定を用いる．各単位時間において，Indirect Tweet の言及先 Tweet が存在するおよび存在しないの 2 値があるとすれば，単位時間あたりのデータの分布はベルヌーイ分布になると考えられる．ベルヌーイ分布の自然共役分布はベータ分布である．そのため，初期状態における事前分布を一様分布と仮定して，事後分布としてベータ分布を求めるものとする．この時，ベータ分布の確率密度関数

$$f(x) = \lambda z^{B-1}(1-z)^{C-1} \tag{21}$$

である．ただし，B, C は正の定数である．この分布の期待値は

$$\mu_r = \frac{B_r}{B_r + C_r} \tag{22}$$

で計算される．B および C の値は式 (23) によって求められる．過去の正解データを表す重複度関数 m を持つ多重集合を A，単位時間を $r = 1, 2, ...h(h$ は任意の自然数) とするとき，

$$B_r = 1 + m(r) \tag{23}$$

$$C_r = 1 + |A| - m(r) \tag{24}$$

である．期待値 μ_r が単位時間に対する関連度となる．

3.6 Tweet 間における総合的な関連度の計算

3.2 節，3.3 節，3.4 節で求めた関連度を合計することで，処理対象 Tweet と計算対象 Tweet 群との関連度を計算する．

$$\mathrm{Sim}(v, p_j) = j_v + s_j + t_j \tag{25}$$

Sim() は計算対象 Tweet 群に属する Tweet p_j とその Tweet をしたユーザ $v \in M_u$ を与えた際に，Tweet 間の総合関連度として，各関連度の総和を返す関数である．ここで，右辺はユーザ間あるいは Tweet 間の関連度を表すものであり，j_v は式 (13) で求められる処理対象 Tweet を投稿したユーザ u とユーザ v のユーザ間関連度であり，s_j は式 (17) で求められる Tweet p_j と p_i の内容的 Tweet 間関連度であり，t_j は式 (22) で求められる p_j と p_i の時間的 Tweet 間関連度である．

3.7 計算対象 Tweet 群の順位付け

計算対象 Tweet 群 $\{p_j \mid p_j = \mathrm{List}(P, p_i, l), \mathrm{Id}(p_j) \in M_u\}$ に属する各 Tweet について，3.6 節で求めた総合関連度の値で降順になるように 1 位から n 位まで順位付けを行う．

4. 関 連 研 究

本研究における Indirect Tweet，すなわちエアリプそのものを主題とした既存研究は国内外を含めほとんど存在しない．「Indirect Tweet」あるいは「Indirect Mention」と名付けた研究の多くは，他のユーザに対するリンクや URL を用いた Tweet や，特定の情報を含む Tweet との対比として Indirect Tweet と表記しているものであり [2] [3]，一般的なユーザの慣習により発生したエアリプそのものには，存在が認知されていながらも深く言及されてはいない [4] [5]．Belkaroui ら [6] は Reply や Mention のみでない特定の話題に関連する Tweet を検索できるようにする必要性を主張しているが，その実験においては一定のキーワードを Tweet 群に設定しており，また，ユーザが意識した暗黙的な関連性でなくハッシュタグや同一 URL によるユーザの意識していない関連性を重視している．そのため，本研究のような投稿者のフォロワーやその一部のユーザに対する Tweet を抽出するのには適していない．

Indirect Tweet と言及先 Tweet では，共通の話題を取り扱っていると考えられる．Twitter のようなマイクロブログを対象とした特定の話題やトピックの抽出を対象とした研究は多数ある．元来，トピック抽出の研究では多量のドキュメントを対象としていた [7]–[11]．同義語や多義語，あるいは表記ゆれという問題に対して，LSA(Latent Semantic Analysis：潜在的意味インデキシング) [12]，ドキュメントからのトピック抽出に LDA(Latent Dirichlet Allocation：潜在的ディリクレ配分法) [13] などが提案され用いられている．しかし，このような手法はドキュメントが一定以上の長さであるときに有効であ

り，マイクロブログの様な Tweet 一つひとつが短文である環境では有効に機能しない事が指摘されている．そのため，Zhao ら [14] による Twitter-LDA モデルや，Steyvers ら [15] による Author-topic モデルが提案された．

時系列文書を取り扱うことになるマイクロブログでは，時間的な情報の変化は，トピックの変遷という観点でも，ユーザの興味の推移という観点でも重要となる．しかしながら，LDA では時間的情報は考慮されない．そのため，佐々木ら [16] による研究では，Twitter-LDA モデルを改良し，時間的情報を取り扱えて，ユーザの興味と話題の時間発展を考慮可能である Twitter-TTM モデルを提案している．一方，このような手法においては，トピックは 1 つのタイムラインを表すのに用いられているか，1 つの Tweet に 1 つのトピックのみであり，Tweet 間が関連することを前提とした Indirect Tweet と言及先 Tweet を的確に関連付けられるとは言い切れない．

Indirect Tweet はある程度親しいあるいは観察対象となりうるユーザ間で投稿されやすい．ユーザ間の親しさを判定する手段としては，従来ユーザプロフィール推定のような研究が主であった．これらの研究では主にフォロー関係によるソーシャルグラフを用いてきた [17] が，この方法ではフォローの理由や活発な交流が為されているかを考慮できないとして，奥谷ら [18] の手法が提案されている．奥村らはメンション情報によるソーシャルグラフの構築を考えた．また，上里ら [19] はこの手法の重要単語の算出方法を変更することにより，上位の概念の属性単語を抽出することに成功している．このように，フォロー関係や Reply を用いることによって同一の属性を持つユーザを推定することは可能だが，本実験では相互フォロー関係であることを前提としている上，Indirect Tweet の対象となるユーザと Reply の対象となるユーザが必ずしも一致しない点からこれらの情報を活用することは難しい．

青島ら [1] は特定の話題に関する Tweet の要約を目的として，単語の共起と時系列的な近さを用いた指標と，出現間隔の同一性に基づく指標から単語間の関連度を算出している．しかし，ここで用いられている時系列的な近さの関数は 5.3 節からも明らかなように Indirect Tweet と言及先 Tweet には適さない．また，この手法は 1 つの大きなイベントを前提としているため，偶発的に発生し相互フォローによる集団の中でも小規模な人数が投稿する Indirect Tweet に適用するのは難しいだろう．

本研究は，Twitter の中でも極めて局所的な話題を対象とし，現実世界でのイベントとの関わりが薄い Tweet について収集することになる．また，Indirect Tweet の特性上相互フォローや頻繁な Mention を収集条件に利用することが良い判断とは言い切れない．そこで，筆者らはユーザの直近の Tweet から得られる関連度や，非明示的かつ 2 者間以上のやり取りであることから Indirect Tweet と言及先 Tweet の時間間隔が短いことを考慮した上で，Tweet 間の関連度を求めている．

5. 評　　価

本実験の目的は，Indirect Tweet から言及先 Tweet を効率的に収集するために，提案手法が計算対象 Tweet 群に対して適切に順位付けを行えるか検証するものである．また，予備実験にて他の手法が適用可能であるかを検討する．

5.1 実 験 条 件

本節では，評価実験の実験条件について述べる．表 1 にて，ある特定のユーザ u からフォロー関係において 1hop 以内で，かつ鍵アカウントでなかったユーザの数を示す．このうち，本実験では相互フォローであるユーザから Tweet を収集する．このような，多くのユーザによる膨大な Tweet 群から，任意の処理対象 Tweet から言及先 Tweet を探し出すことは難しい．

また，Tweet の収集期間は 2016 年 9 月 28 日〜9 月 30 日，2016 年 12 月 14 日〜12 月 18 日，2016 年 12 月 25 日，2017 年 1 月 2 日〜1 月 5 日とする．この期間において，Tweet を人手で確認した．この期間は筆者が日常的に HomeTimeline を閲覧する中で，u が Indirect Tweet を多く投稿していると主観的に予想した日付であり，データ収集の効率以上の意図はない．

実験環境は表 2 の通りである．単位時間を分としているのは，Twitter の公式クライアントが，閲覧時の時刻より 1 時間未満の Tweet であれば分単位で投稿時刻をユーザに提示するからである．厳密には，閲覧時の時刻より 1 分未満の Tweet である場合は秒単位でユーザに提示されることになるが，ここでの差異は無いものとした．

また，各関連度の計算について，次の条件を与える．

（1） 3.2 節のユーザ間関連度の計算では，2016 年 11 月 14 日時点において，$k = 200$ として，Retweet と Quote Tweet に分類される Tweet を除いた Tweet 群を基に計算した

（2） 3.5 節における $|A| = 50$ とした．また，その学習データにおける Indirect Tweet と言及先 Tweet の時間差の分布を図 2 に示す

（3） 処理対象 Tweet $p_i \in \mathrm{I}(P)$ の言及先 Tweet は必ず 2 件以上あるものとする．すなわち，最小で $|\mathrm{R}(p_i, P)| = 2$ である．最大値は $|\mathrm{R}(p_i, P)| = 13$ であった．この条件は処理対象 Tweet の言及先 Tweet が 1 件である場合，その多くは対象とするユーザの名前を本文中に記述しており，ユーザがその言及先 Tweet を探し出すのは容易であると判断したためである．

順位付けの評価は MRR(Mean Reciprocal Rank：平均逆順位) によって行う．MRR は複数の適合文書 (本稿では言及先 Tweet を指す) がある際において用いられる指標であり，適合文書間での適合度の違いは考慮しない．計算式は式 (26) およ

表 1　ユ ー ザ u

フォロー数	2177
フォロワー数	2058
相互フォロー数	1508
相互フォロー内の Tweet 数/日	約 57000

表 2　実 験 環 境

形態素解析機	MeCab 0.996
辞書	mecab-ipadic-NEologd
単位時間	分 (注3)
対象時間	処理対象 Tweet より 60 分前まで
推定対象 Tweet	Tweet，Reply，Mention

び式 (27) の通りである．ここで，a_g は課題において推定された計算対象 Tweet 群における言及先 Tweet の順位である．

$$RR_g = \frac{1}{a_g} \tag{26}$$

$$MRR = \frac{1}{|\mathrm{R}(p_i, P)|} \sum_{g=1}^{k} RR_g \tag{27}$$

本実験においては，言及先 Tweet が 2 つ以上存在するため，必ず MRR<= 0.75 となり，$|\mathrm{R}(p_i, P)|$ の数が大きくなるほど MRR の最大値は小さくなる．そのため MRR の最大値が 1 となるように，言及先 Tweet の数 $|\mathrm{R}(p_i, P)|$ に応じて，式 (27) の正規化を行う必要がある．実験結果では MRR を正規化した値で記す．正規化に用いる値は式 (28) で求めるものとする．

$$\mathrm{Max}(p_i, P) = \frac{1}{|\mathrm{R}(p_i, P)|} \sum_{w=1}^{k} \frac{1}{w} \tag{28}$$

5.2 予 備 実 験

5.3 Indirect Tweet と言及先 Tweet の時間差

本実験環境における Indirect Tweet と言及先 Tweet の組み合わせ 425 組について，その時間差の頻度を図 3 で示す．Indirect Tweet に複数の言及先 Tweet が存在する場合，時間差が最小となる組み合わせのみを対象としている．

グラフより，Indirect Tweet の 60%程度は，言及先 Tweet より 2 分以内に行われていることが分かる．また，1 分以降は時間が経過する毎に Tweet 頻度は指数関数的に減少する．しかし，21 分の組み合わせなど場合によっては Twitter 上の機能は用いずユーザ名などを本文中に記述し，コンテキストの依存を軽減しながら，対象とするユーザにとって明示的な Indirect Tweet を投稿する状況も見受けられた．

この結果より，0 分以内と 1 分以内の頻度がほぼ同等であることから，単純に時系列順で関連度を決定するような手法は適切でないことが分かる．

5.4 Indirect Tweet と Reply および Mention の関係

5.1 節の条件において，ユーザ u から他のユーザに対する Indirect Tweet と Reply および Mention において，重複が存在するか，対象となるユーザの規模はどの程度かを検証した．結果を表 3 に示す．u が Indirect Tweet の対象としたユーザ数は 38 人であり，Reply および Mention の対象としたユーザ数は 79 人であった．その内，どちらの対象ともなっているユーザは 26 人であり，u のコミュニケーションの対象となったユーザ全体からすると 22%のみである．

この結果より，Indirect Tweet と Reply，Mention の対象となるユーザは約 8 割が重複せず，Reply や Mention をもとに，クラスタを判定するような手法は適用できないことが分かる．

5.5 実　　験

次に示す各手法について実験を行った．ただし，すべての式について，計算対象 Tweet 群 $\{p_j \mid p_j = \mathrm{List}(P, p_i, l), \mathrm{Id}(p_j) \in M_u\}$ であり，M_u に属する任意のユーザを v とする．

（1） 提案手法

3 章で解説した提案手法を用いる．文字情報を用いたユーザ間関連度と内容的 Tweet 間関連度と時間的距離を用いた時間的 Tweet 間関連度の和が順位付けに用いられる関連度となる ((25) 式参照)．以降，これを総合関連度と呼ぶ．

（2） 提案手法 (時間的 Tweet 間関連度を正規化した場合)

提案手法の内，時間的距離 Tweet 間関連度の値を，最大値が 1 となるように正規化したものである．

$$\mathrm{NormSim}(v, p_j) = j_v + s_j + t'_j \tag{29}$$

ただし，t'_j は各関連度計算の課題における t_j の値を，最大値で除算したものである．以降これを正規化総合関連度と呼ぶ．

（3） 文字情報による関連度のみを用いた場合

文字情報によって計算されるユーザ間関連度 (3.2 節) と内容的 Tweet 間関連度 (3.3 節) の和によって順位付けを行う．以降，これを文字情報関連度と呼ぶ．

$$\mathrm{CharSim}(v, p_j) = j_v + s_j \tag{30}$$

（4） 時間的距離のみを用いた場合

処理対象 Tweet からの時間的距離のみによって順位付けを行う手法．言及先 Tweet 候補が処理対象 Tweet に時間的に近

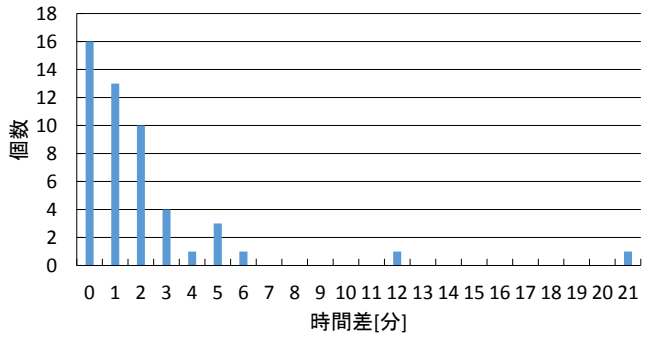

図 2　学習に用いた Indirect Tweet と言及先 Tweet の時間差の分布

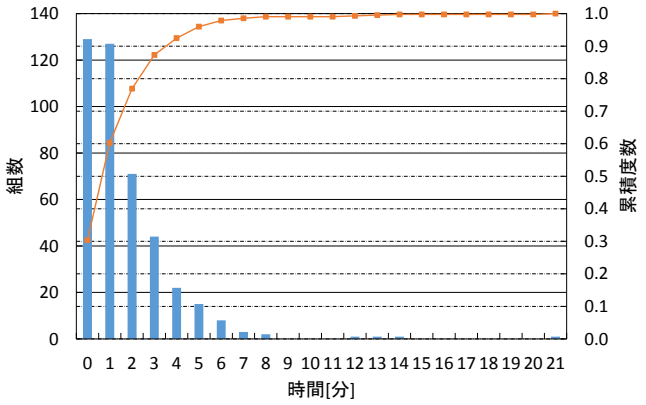

図 3　425 組の Indirect Tweet と言及先 Tweet の時間差における頻度

表 3　u のコミュニケーション対象ユーザ

Indirect Tweet のユーザ数	38
Reply および Mention のユーザ数	79
合計ユーザ数	117
重複ユーザ数	26
重複率	22%

ければ近いほど順位が高くなる．単位時間による計算ではないことに留意されたい．今回の場合は秒単位で計算している．$p_i \in \mathrm{I(P)}$ として式 (31) で表され，これを時系列関連度と呼ぶ．

$$\mathrm{TimeDefSim}(p_j) = |\mathrm{Time}(p_i) - \mathrm{Time}(p_j)| \tag{31}$$

5.6 実験結果

実験を行った結果を図 4 で示す．横軸は課題番号を表している．課題番号と言及先 Tweet の数 $|\mathrm{R}(p_i, P)|$ の対応関係を表 5 で示す．縦軸は正規化された MRR の値であり，値が高いほど良い結果を得たことになる．また，言及先 Tweet 数ごとの正規化された MRR の平均値を図 5 で示す．また，MRR の平均値は表 4 の通りである．

5.7 考察

表 4 より，正規化総合関連度が MRR において最も良い結果を得た．また，文字情報関連度よりも総合関連度の結果が良いことから，文字情報のみを用いるのではなく，時系列情報も用いた方が良いことは明らかである．しかし，時系列情報のみを用いた時系列関連度の MRR は最も低い値となっている．これは，実験の対象とした，相互フォローによって構成される集団のユーザ数が極めて多く，全体の Tweet 頻度が高いことに起因するものだと考えられる．また，本実験の定義では 1 つの処理対象 Tweet に複数の言及先 Tweet が存在することから，処理対象 Tweet と言及先 Tweet の時間差が必ずしも小さくないことも原因と言えるだろう．

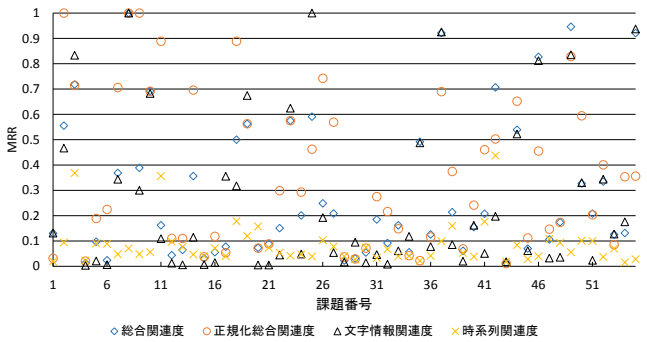

図 4　実験結果 (課題数=55)

表 4　各手法における MRR

	平均	標準偏差
総合関連度	0.29	0.28
正規化総合関連度	0.36	0.30
文字情報関連度	0.25	0.31
時系列関連度	0.08	0.08

表 5　課題番号と $|\mathrm{R}(p_i, P)|$ の対応関係

課題番号	言及先 Tweet 数 $|\mathrm{R}(p_i, P)|$
1〜18	2
19〜33	3
34〜36	4
37〜40	5
41〜46	6
47〜50	7
51〜52	8
53	10
54	12
55	13

平均値では正規化総合関連度が良い結果となったが，図 4 を見れば分かるように，全ての課題に対して有用であるわけではない．正規化総合関連度はグラフで言えば右側，つまり $|\mathrm{R}(p_i, P)|$ の値が大きく言及先 Tweet の数が多いほど MRR の値は小さくなる傾向があるように見える．ここで，図 5 に着目する．言及先 Tweet の数ごとに MRR の平均値を計算した場合，正規化総合関連度において最も良い値を得たのは，言及先 Tweet 数が 2 件だった場合である．また，言及先 Tweet 数が増えた場合では，7,8,12 件の場合においては他の手法よりも良い結果となっている．言及先 Tweet が 7 件以上かつ正規化総合関連度が他の手法に比べて良い値を得た場合の順位付けを比較すると，その多くは，複数ある言及先 Tweet の内の 1 件を 1 位に順位付けしていた．また，その 1 件以外の順位付けについては正規化総合関連度が，言及先 Tweet を他の手法よりも低い順位にしている場面が多く見受けられた．

これらの結果から，他の手法に比べて正規化総合関連度は，複数存在する言及先 Tweet に対して，処理対象 Tweet から (学習データにもよるが今回の場合は) 時系列上において近しい 1 件の言及先 Tweet を適切に上位に順位付けすることが示唆された．しかし，話題が持続的に発生している場合には他の手法に劣ることもあり，話題の全体を収集するのには向いていない可能性がある．他の手法との違いは，いわば時間的距離を用いた関連度に対する重み付けの違いである．表 4 からすれば，文字情報関連度に対して総合関連度は 0.4 程度値がよく，総合関連度に対して正規化総合関連度は 0.7 程度値がよい．また，標準偏差については総合関連度が最小となっている．適切な重み付けにより，更なる改善につながる可能性はあるだろう．しかし，正規化総合関連度，総合関連度，文字情報関連度のいずれの手法においても課題ごとに優劣が前後していることから，この重み付けの適切な値を求めることは難しいと考えられる．なぜなら，正確な言及先 Tweet の数は分からず，特定の話題に関連する Tweet がどの程度の時間行われるのかについては不明だからである．そのため，本提案手法の関連度をパラメータとして変動させることで順位付けを改善するのは難しいだろう．

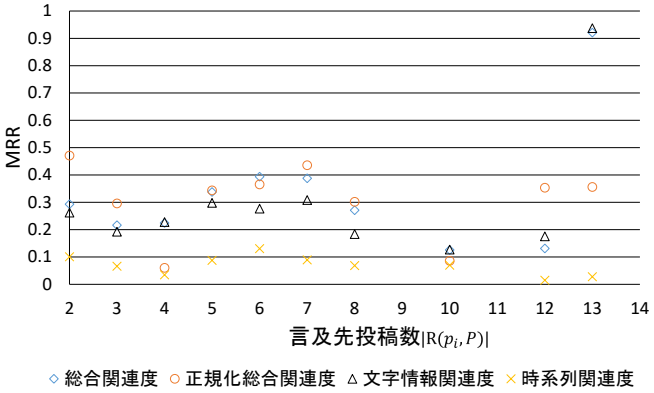

図 5　実験結果 (言及先 Tweet 数ごとの平均値)

一方で，正規化総合関連度は殆どの課題に対して上位 10 件以内に 1 つ以上の言及先 Tweet を順位付け，上位 30 件以内であれば 4 つの課題を除いて 1 つ以上の言及先 Tweet を順位付けしている．結果から，本手法はユーザに対して少量に削減した計算対象 Tweet 群を提示するのに向いていると言える．

また，今回ベイズ推定による学習に用いた Indirect Tweet と言及先 Tweet の時間差は，1 つの Indirect Tweet に対して時間差が最小である 1 つの言及先 Tweet のみを設定している．そのため，もし言及先 Tweet 一つひとつに対して時間差を求めたのであればまた違う結果が得られる可能性がある．その場合，処理対象 Tweet から時間的に離れていた言及先 Tweet がより上位に順位付けされることになるだろう．しかし，今回の実験環境のように処理対象 Tweet に対して時間軸上に一定の間隔で言及先 Tweet が存在しているのであれば，処理対象 Tweet との時間差が最小である言及先 Tweet から次の言及先 Tweet を推定する手法が有効だと考えられる．

ゆえに，更なる改善案としては，本手法によって上位に順位付けされた少量の計算対象 Tweet 群をユーザに提示し，その内からユーザが正解となる言及先 Tweet を選択し，他の言及先 Tweet を収集するような疑似適合フィードバックによる手法が極めて有用だと考えられる．Twitter における疑似適合フィードバックの先例としては宮西ら [20] の研究が挙げられる．

6. おわりに

本論文では，ユーザの慣習によって発生した Indirect Tweet という，Tweet あるいはユーザを暗黙的な言及先とする Tweet から，それに関連付けられる言及先 Tweet を効率的に収集するために，2 つの Tweet 間で関連度を算出し順位付けをする手法を提案した．関連度には 3 つの指標が用いられている．提案手法では，ユーザの嗜好および Tweet 間の単語での同一性を文字情報によって考慮している．また，Indirect Tweet と言及先 Tweet の時間差に見られる偏りをベイズ推定によって学習することにより Indirect Tweet の特性を利用し，Indirect Tweet が単純な応答である場合文字情報のみでは下位になってしまう現象を軽減することが出来た．提案手法において，時系列情報に重み付けを行うことによって順位付けの精度が向上することが示唆されているが，課題によって適した重みが変わることから根本的な解決にはならないだろう．

今回の実験で対象とした集団では，Indirect Tweet が言及先 Tweet を指し示す際に，指示代名詞を用いられていることは少なかった．またユーザ名や Tweet 内で意味的に共通する単語を用いる場面が多く見受けられた．このような Indirect Tweet の特性が時間差の偏りも含めて他のユーザの集まりでも見られる現象であるのかは引き続き調査が必要である．

文　献

[1] 青島傳隼, 坂本翼, 横山昌平, 福田直樹, 石川博. 「文脈的なつながりを考慮したツイート群の効果的な抽出・提示手法の実現」. 情報処理学会論文誌データベース（TOD）. 2013, vol. 6, pp. 61-84.

[2] IMRAN Muhammad, et al. “Extracting information nuggets from disaster-related messages in social media.” Proc. of ISCRAM, Baden-Baden, Germany, 2013.

[3] TANAKA Yuko, SAKAMOTO Yasuaki, HONDA Hidehito. “The impact of posting URLs in disaster-related tweets on rumor spreading behavior.” System Sciences (HICSS), 2014 47th Hawaii International Conference on. IEEE, 2014. pp. 520-529.

[4] SATYANARAYAN Ashwin, DAS Bk Sarthak, KRISHNAN Divya. “Analyzing Advertisements on Twitter during Valentine ’ s Month.”

[5] ISBISTER Joseph. “Exploring adolescents use of social networking sites and their perceptions of this can influence their peer relationships.” 2013. PhD Thesis. Institute of Education, University of London.

[6] BELKAROUI Rami, FAIZ Rim, ELKHLIFI Aymen. “Using social conversational context for detecting users interactions on microblogging sites.” Revue des Nouvelles Technologies de l’Information, 2015.

[7] 水田昌孝，熊野雅仁，小野景子，木村昌弘．「文書ストリームからのバースト潜在トピック抽出における t-LDA 法の性能検証」，情報処理学会研究報告．MPS，数理モデル化と問題解決研究報告 2010-MPS-81(10)，pp. 1-6，2010-12-09.

[8] 橋本泰一，村上浩司，乾孝司，内海和夫，石川正道．「文書クラスタリングによるトピック抽出および課題発見」，社会技術研究論文集，5，pp. 216-226，2008.

[9] 木村学，斉藤和巳，上田修功．「確率モデルに基づく文書ストリームからのホットトピック抽出の一検討」，電子情報通信学会技術研究報告．AI，人工知能と知識処理 106(38)，pp. 51-56，2006-05-11.

[10] 高橋祐介，横本大輔，宇津呂武仁，吉岡真治．「ニュースにおけるトピックのバースト特性の分析」，情報処理学会研究報告．自然言語処理研究会報告 2011-NL-204(6)，pp. 1-6，2011-11-14.

[11] 余東明，石川孝．「コミュニティウェブにおけるアクティブ情報検索ためのトピック抽出」，人工知能学会全国大会論文集 JSA103(0)，pp. 68-68，2003.

[12] Scott Deerwester, Susan T.Dumais, George W.Furnas, Thomas K. Landauer, Richard Harshman, “Indexing by latent semantic analysis,” Journal of the American Society of information Science, 41(6), pp. 391-407, 1990.

[13] Blei, D.M., Ng, A.Y. and Jordan, M, “Latent Dirichlet allocation,” The Journal of Machine Learning Research, Vol.3, pp. 993-1022, 2003.

[14] W.X. Zhao, J.jiang, J.He, Y.Song, P.Achananuparp, E.-P. Lim, and X.Li, “Topical keyphrase extraction from twitter,” The Annual Meeting of the Association for Computational Linguistics 2011, pp. 379-388, 2011.

[15] M.Steyvers, P.Smyth, M.Rosen-Zvi, and T.Griffiths, “Probabilistic author-topic models for information discovery,” SIGKDD2004, 2004.

[16] 佐々木謙太朗，吉川大弘，古橋武．「Twitter におけるユーザの興味と話題の時間発展を考慮したオンライン学習可能なトピックモデルの提案」，情報処理学会論文誌．数理モデル化と応用 7(1)，pp. 53-60，2014-03-28.

[17] 岡本大輝, 豊田正史, 喜連川優．「マイクロブログにおける対話ネットワークと Tweet 内容を併用したユーザ推薦に関する一考察」. 電子情報通信学会技術研究報告. DE, データ工学. 2013, vol. 113, pp. 169-173.

[18] 奥谷貴志, 山名早人.「メンション情報を利用した Twitter ユーザプロフィール推定」. 研究報告情報基礎とアクセス技術（IFAT）. DEIM Forum 2014.

[19] 上里和也, 田中正浩, 浅井洋樹, 山名早人．「メンション情報を利用した Twitter ユーザプロフィール推定における単語重要度算出手法の考察」. 研究報告情報基礎とアクセス技術（IFAT）. 2014, vol. 2014, pp. 1-6.

[20] 宮西大樹，関和広，上原邦昭．「マイクロブログ文書の選択による適合フィードバックを用いた疑似適合フィードバックの検索性能改善」，情報処理学会論文誌 55(5)，pp. 1585-1594，2014-05-15.