ブログのフレンドマイニングにおける類似度計算法の比較に関する研究

M2011MM027 伊藤公郎
指導教員：石崎文雄

1 はじめに

手軽に個人の情報を発信でき，他人とのコミュニケーションが取れるもとのしてソーシャル・ネットワーキング・サービス（SNS）が注目を集めている．
SNS の中でも平成 15 年頃から利用者が急増し現在も安定して多く利用しているブログがある．ブログで代表的なものに，FC2 ブログやライブドアブログ，アメーバブログなどがあり，また，ブログ機能がある SNS として mixi などがある．これらの代表的なブログの利用者数は，2011 年 6 月の調査では FC2 ブログは約 560 万人，ライブドアブログは約 430 万人，アメーバブログは約 1645 万人で現在も増加傾向にある．ブログを対象にした研究には，ブログ記事やタグから類似するブロガーを発見するフレンドマイニングや話題抽出などがある．フレンドマイニングに関する研究は，ブログ以外にも SNS に対しても行なわれている．

本研究では，インターネット上で多くの人が自由に個人の趣味・嗜好など興味があるもの・ことを掲載でき，それを他人が閲覧できることで興味の共有を出来るブログに注目し，ブログの掲載記事やタグなどのデータの内容に基づいてまだ知らないが共通の興味を持つブロガーつまり，潜在的な友人を発見しそれぞれ推薦するための方法を考える．その中でも特に，ブログのデータから共通の興味を持つ潜在的な友人かを判断するのに使用する類似度に焦点を当てる．類似度を計算する方法としてよく使用されているコサイン類似度のほかにピアソン相関係数や偏差パターン類似度，ジャカード係数，ダイス係数，シンプソン係数を用いて類似度をそれぞれ計算する．
異なる類似度計算で出た結果の性能評価をして比較し，どの類似度計算方法がブログでの潜在的な友人を推薦する場合に最も有効なのかを検証する．

2 SNS の類似度によるデータマイニングに関連する研究

2.1 Blog のフレンドマイニングに関する先行研究

Nitin Agarwal ら [1] の研究は，ブログのタグとスニペットの類似度から類似するブロガーの発見する研究である．この論文の提案する手法の手順は，まずブロガーに関連するカテゴリを見つける．
次にブログのサイトレベルのタグと投稿記事ごとのタグ，ブログ記事のスニペットからプロファイルを構築し，LSA による高次元の特徴ベクトルを概念空間に転換し，プロファイルに基づきコサイン類似度により類似ブロガーを発見するものでこの探索アルゴリズムを CWS(Collective Wisdom based Search) と呼ぶ．

Dou Shen ら [2] の研究は，ブログのエントリに基づきブロガー間の潜在的友人を発見するフレンドマイニングをする研究である．潜在的友人は，自分のブログと同様のトピック分布を共有するブロガーと定義している．
この論文では，ブロガー同士互いに知り合いでなくても共通の興味を持つ潜在的友人を発見するための三つの手法を提案している．最初のものは，コサイン類似度ベース法と呼ばれる，ブログの内容間のコサイン類似度を計算することによってブロガー間の類似度を決定する．
2 番目は，トピックベース手法として知られており，潜在トピックモデルを用いて潜在的トピックの発見に基づき，トピックレベルで類似度を計算する．
3 番目は，2 つのレベルの類似性に基づき 2 段階で実施する．最初の段階では，ブロガーのためのトピック分布を構築するために利用されている 既存のトピック階層を用いる．その後，第二段階で，詳細な類似性の比較を最初の段階で発見されて互いに関心の近いブロガーで行われる．

Panagiotis Symeonidis ら [3] の研究は，オンラインソーシャルネットワーク (OSN) で共通の友人の数に基づいてユーザに新たな友人を紹介する研究である．彼らは，FriendTNS アルゴリズムを提案している．
最初のアプローチは，ノードの構造を中心に，ネットワークの局所特徴に基づいている．
第二のアプローチは，ネットワーク全体のパス構造を検出し，グローバルな機能に基づいている．FriendTNS アルゴリズムが精度の面で他のアプローチよりも優れていて，効率的時間であることを示している．

2.2 先行研究との相違点

ブログのフレンドマイニング（潜在的友人の発見）の先行研究では類似度計算法を用いたフレンドマイニングを行なっているが，そのほとんどがシンプルでわかりやすいコサイン類似度を用いた研究・実験を行なっている．コサイン類似度を用いたフレンドマイニングと彼らが提案する手法との比較で別の類似度計算法を用いたフレンドマイニングの性能・精度の比較などは述べられていない．類似度計算法を用いたフレンドマイニングでコサイン類似度の場合が最も性能・精度が良いか分からない．

そこで本研究では，Ameba blog のデータを使用して，ブログの投稿記事からプロファイルを構築し，複数の類似度計算のアルゴリズムを使用して類似ブロガーを発見し，コサイン類似度を用いたフレンドマイニングと別の類似度計算法を用いたフレンドマイニングを複数個行い性能・精度を比較・評価してコサイン類似度を用いた場合よりも性能・精度が高い類似度計算法を用いたフレンドマイニングを発見することが目的である．

3　ブログのフレンドマイニングのシステム構築

3.1　ブログの類似度計算までの流れ

まず，ブログの類似度計算までの流れを図 1 に示す．図 1 の流れを説明する．

(1) まず，Ameba Blog のデータを収集する．データ収集のためのクローラに GNU Wget を用いる．GNU Wget を用いて Ameba 人気ブログランキング 2012 年度総合ランキングの上位数名からクロールをはじめてデータを収集する．

(2) 次に，データの変換を行なう．GNU Wget で収集した Ameba ブログのデータはファイル形式が HTML ファイルなのでデータ解析できるファイル形式に変換する必要がある．本研究では，HTML ファイルを TEXT ファイルへ変換するソフト HtoX32c を用いて TEXT ファイルへ変換する．

(3)TEXT ファイルへ変換したブログデータを統計ソフト R へ読み込み，読み込んだブログデータを形態素解析により単語に分ける．形態素解析は，R のパッケージソフト RMeCab を用いる．RMeCab は，R から形態素解析ソフト和布蕪（MeCab）を呼び出して使うインターフェイスである．RMeCab を用いて TEXT ファイルから単語を取り出す．本研究では，品詞が名詞で品詞細分類が一般，固有名詞，形容動詞語幹を使用するためその単語のみを取り出しファイルに保存する．この操作を収集したブロガーごとのフォルダと全てのファイルを合わせたフォルダに行なう．

(4) 全てのファイルから取り出された名詞（一般，固有名詞，形容動詞語幹）とブロガーごとに取り出された名詞（一般，固有名詞，形容動詞語幹）で同じ単語がある場合は，”1”を，ない場合は，”0”を与えたファイルを作成する．

(5) ブロガーごとに作成した”1”，”0”のファイルを用いて類似度計算をする．本研究では，コサイン類似度とジャカード係数，シンプソン係数，ピアソン相関係数，偏差パターン類似度，ダイス係数の 6 つの類似度計算を行なう．それぞれの類似度計算を R で関数としてプログラムを作成する．表示結果は全ての列のコサイン尺度の行列を表示する．

3.2　使用するブログデータ

本研究では，Ameba Blog のブログデータを用いる．国内での人気ブログサービスの 2012 年のアクティブユーザー数を表 2 に示す．

表 1 から，Ameba blog のアクティブユーザーは，40 万人で，10 万人の FC2 ブログや 9 万人の Yahoo!ブログ，1 万人以下のライブドアブログ，シーサーブログ，ジュゲムブログ，ココログ，ヤプログ，楽天ブログ，はてなブログ，goo ブログのブログサービスと比べて圧倒的にアクティブユーザーが多い．アクティブユーザーが多いのでより新しい情報でブロガー間の類似度比較ができ，現在のブロガーの趣味嗜好で潜在的友人を推薦できるのではないかと考える．以上の理由から本研究では，Ameba blog のブログデータを使用する．

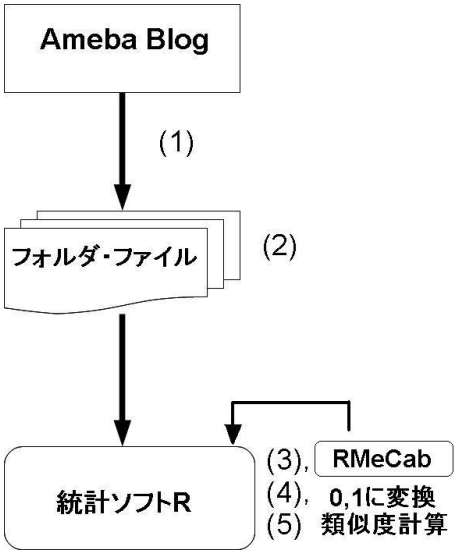

図 1　ブログの類似度計算までの流れ

表 1　2012 年ブログサービス

アクティブユーザ数	ブログサービスサイト名
40 万人	Ameba blog
10 万人	FC2 ブログ
9 万人	Yahoo!ブログ
1 万人以下	ライブドアブログ／シーサーブログ／ジュゲムブログ
1 万人以下	ココログ／ヤプログ
1 万人以下	楽天ブログ
1 万人以下	はてなブログ
1 万人以下	goo ブログ

ブログデータは，Ameba 人気ブログランキング　2012 年度　月間総合ランキングの上位数名からクロールしたものを用いる．表 2 に収集した Blog データの情報を示す．使用するフォルダ数は，ブロガーごとに作られたフォルダに 2 つ以上のファイルが含まれたフォルダで 2 つ以上のファイルが含まれるフォルダのみを使用する．

表 2 収集した Ameba Blog のデータの情報

使用するフォルダ数	使用するファイル数
500	6455

4　R での類似度計算

本研究では統計ソフト R を用いて類似度計算を行なう．本研究で扱う類似度計算は，コサイン類似度，ジャカード係数，シンプソン係数，ピアソン相関係数，偏差パターン類似度，ダイス係の 6 つである．

類似度の値は”0”から”1”の間を取り”1”に近いほど類似度が高いことを示す．

6 つの類似度計算は R 上に関数として作成し計算する．

本研究の類似度計算で使用するブロガーデータは収集した 500 ブロガーで計算する．

4.1 類似度計算プログラム

統計ソフト R で変換した ”0，1 ”ファイルから類似度計算するためにそれぞれ類似度計算するプログラムを関数として作成する．本研究では，コサイン類似度とジャカード係数，シンプソン係数数，ピアソン相関係数，偏差パターン類似度，ダイス係の 6 つの類似度計算法を扱う．6 つの類似度計算法を説明する．

コサイン類似度は，ベクトルの向きの近さを類似性の指標としたもの．

$$sim = \frac{x \cdot y}{(|x| * |y|)}$$

ジャカード係数は，集合 X と Y の共通要素数を少なくとも 1 方にある要素の総数で割ったもの．

$$sim = \frac{x \cdot y}{\Sigma x_i + \Sigma y_i - x * y}$$

シンプソン係数は，集合 X と Y の共通要素数を各集合の要素数の最小値で割ったもの．

$$sim = \frac{x \cdot y}{\min(\Sigma x_i, \Sigma y_i)}$$

ピアソン相関係数は，x と y を 2 つの変数と考え，次元ごとの値の組 (xi,yi) (i=1,2,..,n) の相関係数として算出する．ベクトル x の次元要素 xi (i=1,2,..,n) の平均を mx とし，ベクトル v=x-mx = (x1-mx, x2-mx, .., xn-mx) とおく．同様にベクトル y に対して，w=y-my=(y1-my, y2-my, .., yn-my) とおく．

$$sim = \frac{v \cdot w}{(|v| * |w|)}$$

偏差パターン類似度は，全ベクトルの平均ベクトルからの偏差ベクトルを使う．第 i 次元要素 xi の平均を mi とし平均ベクトルを m=(m1,m2,..,mn) とする．x,y の偏差ベクトルを v=x-m = (x1-m1, x2-m2, .., xn-mn)，w=y-m=(y1-m1, y2-m2, .., yn-m3) とする．

$$sim = \frac{v \cdot w}{(|v| * |w|)}$$

ダイス係数は，集合 X と Y の共通要素数を各集合の要素数の平均で割ったもの．

$$sim = \frac{2 * x \cdot y}{\Sigma x_i + \Sigma y_i}$$

次に類似度計算法のプログラムの一部としてコサイン類似度のプログラムを説明する．
コサイン類似度のプログラムの説明として，予め x にブロガーごとの ”0，1 ”に変換したブログデータを読み込んでおく．
1 行目では，cosine.function で実行できる関数を以下で指定する．2 行目では，x の列数を行と列にもつ行列の row.similarity を作成する．3 から 6 行目では，コサイン類似度計算を行う．3 行目では，i に 1 から x の列に格納されている単語数まで繰り返す．4 行目では，j に 1 から x の列に格納されている単語数まで繰り返す．
5 列目では，row.similarity[i,j] に 2 ブロガーのコサイン類似度を計算する．8,9 行目では，row.similarity の rownames，colnames に x の rownames を与える．
以上の操作によりコサイン類似度を行う．

cosine.function の関数のプログラム

```
cosine.function ← function(x){
row.similarity
← matrix(0, ncol = nrow(x), nrow = nrow(x))
for(i in 1:nrow(x)){
for(j in 1:nrow(x)){
row.similarity[i,j] ←
(x[i,]%*%x[j,])/(sqrt(sum(x[i,]^2))*sqrt(sum(x[j,]^2)))
} }
rownames(row.similarity)
← colnames(row.similarity) ← rownames(x)
return(row.similarity)
}
```

4.2 類似度のグラフ化

6 つの類似度計算法での類似度の値を（類似度の値-平均)/標準偏差 で標準化して重ねてグラフにした．6 つの類似度計算法での類似度の値をグラフ化したものを図 2 に示す．

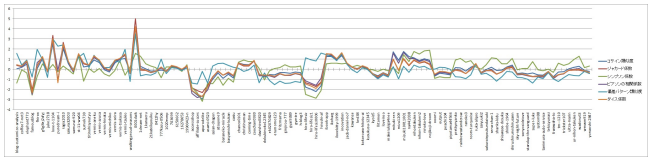

図 2　6 つの類似度計算法での類似度

このグラフから，シンプソン係数と偏差パターン類似度は他の 4 個のグラフと形が大きく違っている．シンプソン係数と偏差パターン類似度は類似度の値が片方が大きい時にもう片方が小さくて，小さい時に大きくなるような対照的な値を取っていた．他のコサイン類似度とジャカード係数，ピアソンの相関係数，ダイス係数は似たグラフの形になった．また標準化した後の類似度の値の差は，コサイン類似度が 7.056778735，ジャカード係数が 7.330443876，シンプソン係数が 5.068154638，ピアソンの相関係数が 6.923488097，偏差パターン類似度が 5.032169851，ダイス係数が 6.92405418 であり，コサイン類似度とジャカード係数が広い範囲に類似度の値を取り偏りが少ないと考えられる．逆にシンプソン係数と偏差パターン類似度は狭い範囲で類似度の値が取られており値に偏りがあると考えられる．

5　アンケートによる比較と性能評価

この節では，実験結果の一部からアンケートを取り，比較と性能評価をしていく．実験結果のデータを使用してある特定のブロガーを 5 ブロガーランダムに選び，その 5 ブロガーとのブログ記事の類似度でアンケートを取る．そのアンケート結果を元に比較と性能評価をする．

5.1 アンケート

500 名のブロガーからランダムで 5 ブロガーを選び本研究のシステムによる 6 つの類似度計算法による類似度の値をそれぞれ出し，類似度の値が高い 15 名を取り出しブログの内容が似ているかを人の手によってのアンケートを取る．アンケートは 8 名に取る．以下にアンケートの内容を示す

・ランダムに選ばれた 5 名のブログとコサイン類似度，ジャガード係数，シンプソン係数，ピアソンの相関係数，偏差パターン類似度，ダイス係数での類似度計算で類似度の値が高い 15 名のブログの一部を読んでもらう．

・ランダムに選ばれた 5 名のブログと類似度の値が高い 15 名のブログを読み，点数をつけてもらう．

・点数は，似ている場合に 1.0 点，一部似ている場合に 0.5 点，似ていない場合 0 点をそれぞれつけてもらう．

5.2 比較と性能評価

本研究では，客観的にブロガー同士の類似性の精度を測る方法がない．そこで本研究では，人の手によるアンケートを取り，そのアンケート結果を用いて独自の方法で精度 (P) を出す主観的な方法での精度の比較と性能評価をする．

本研究での精度 (P) の求め方は以下のように求める．

$$P = \frac{LatentFriend}{N}$$

Latent Friend は，アンケートの点数の合計を表す．N は，似ているか比較するブロガーの数を表す．

アンケートでの点数を元に本研究での精度 (P) を求める．類似度のアンケート結果から上位 5 ブロガー，上位 10 ブロガー，上位 15 ブロガーごとに本研究で用いる精度 (P) を出したものの平均精度を以下の表 3 に示す．

表 3 アンケートによる結果精度 (P) の値

類似度	平均精度
コサイン類似度	0.460
ジャカード係数	0.539
シンプソン係数	0.365
ピアソン相関係数	0.528
偏差パターン類似度	0.370
ダイス係数	0.448

アンケートの点数からの類似度計算法ごとの平均精度では，コサイン類似度の精度 0.460 よりも高い精度が出たのはジャカード係数の 0.539 とピアソン相関係数の 0.528 であり，ダイス係数の 0.448 は同じくらいの精度になり，シンプソン係数と偏差パターン類似度は低い精度となった．

6 まとめ

Ameba Blog のブログデータ 500 ブロガーでコサイン類似度，ジャガード係数，シンプソン係数，ピアソンの相関係数，偏差パターン類似度，ダイス係数の 6 つの類似度計算法による類似度計算を行なって，類似度の値が高いブロガーから潜在的な友人となるブロガーを発見するために類似度の精度の高い類似度計算法はどの類似度計算法かを本章では考察してきた．類似度の値の範囲については，類似度の値の最大値と最小値の差が類似度の値の範囲として範囲が広いものが偏りが少ないと考える．ピアソンの相関係数と偏差パターン類似度が範囲が 0.85 以上で，コサイン類似度とジャガード係数，シンプソン係数，ダイス係数は 0.6 前後であるが，ピアソンの相関係数と偏差パターン類似度はとり得る範囲は− 1.0 から 1.0 で他の 4 つは 0 から 1.0 なのでこのままでは比較できないので標準化したあとに類似度の値の範囲を比較した．コサイン類似度とジャカード係数は範囲が 0.7 以上で，ピアソンの相関係数とダイス係数は 0.6 以上で，シンプソン係数と偏差パターン類似度は 0.5 以上でコサイン類似度とジャカード係数が 6 つの中では偏りが少なく類似度の値が取られている．精度については，ジャカード係数とピアソンの相関係数がコサイン類似度より高い精度が出ている．シンプソン係数と偏差パターン類似度は，他の 4 つの類似度計算法より低い精度となった．総合的にコサイン類似度より性能がいいと考えるのは類似度の値の範囲でも本研究での精度でも 6 つの類似度計算法の中で最も良い結果が出たジャカード係数である．よって本研究では，ジャカード係数がブログの類似度計算によるフレンドマイニングをする場合に適していると考える．

7 今後の課題

本研究では，使用したブログのブロガーの数が 500 ブロガーだったが，収集数を広げることでもっと類似度の精度に差が出た．また，本研究で収集したブログサイトが Ameba Blog だけだったが別のブログサイトからのブログも使用することで精度が高くなると考える．ブログから形態素解析で取り出した重要単語が名詞だけなので前後関係がわかる形容詞や形容動詞，また動詞などからも類似度計算出来れば精度が上がると考える．今後は，以上のような改良点を課題としてフレンドマイニングをしていくことが重要になってくると考える．

参考文献

[1] Dou Shen, Jian-Tao Sun, Qiang Yang, Zheng Chen, “ Latent Friend Mining from Blog Data, ”Department of Computer Science and Engineering Hong Kong University of Science and Technology, Hong Kong, 2006.

[2] Nitin Agarwal, Huan Liu, Shankara Subramanya, John J. Salerno and Philip S. Yu, “ Connecting Sparsely Distributed Similar Bloggers, ”University of Arkansas at Little Rock, Little Rock, AR 72204, 2009.

[3] Panagiotis Symeonidis, Eleftherios Tiakas, Yannis Manolopoulos, “ Transitive Node Similarity for Link Prediction in Social Networks with Positive and Negative Links, ”Proceedings of the 4th ACM Conference on Recommender Systems (ACM/RECSYS), pp.183-190, Barcelona, Spain, 2010.