

深層学習を用いた Twitter 利用者のトピックに対する興味傾向推定

Estimating Twitter Users' Interest Tendency for Topics Using Deep Learning

石川 理一朗† Riichiro Ishikawa
山本 雄平‡ Yuhei Yamamoto
佐野 睦夫† Mutsuo Sano

1. はじめに

現在、社会は情報化が進み、情報過多の状態となっている。2018 年に実施された世論調査[1]では「今の社会は情報が多すぎる」という人が全体で 8 割を超えていた。情報過多の状態はマーケティングにおいて、適切な情報を得られない利用者だけでなく情報を提供したい企業にとっても問題となる。Amazon.com[2]の通販を例とすると、工具のドライバーを検索した場合[3]、類似する商品の情報が 3 万件以上[3]提示される。この問題に対して Amazon は、単純な売り上げ数や値段によるフィルタリングの他に利用者の購買履歴等の情報を基に利用者の傾向を分析し推薦を行うシステム[4]を導入することで、膨大な商品情報の中から利用者に適した商品情報の提示を実現している。

利用者から得られる情報を基にした推薦システムは様々な分野で導入や開発が行われており、その精度の向上や推薦手法に注目が集まっている。マイクロブログの一種である Twitter にも推薦システムが存在し、利用者の興味に合致するトピックや利用者の推薦、広告の最適化等の形で活用されている。しかし、Twitter の推薦アルゴリズムはアドレス帳や過去のフォローのパターンなど多数の判断材料に基づいて推薦を行っているが推薦されるアカウントが必ずしも興味のある対象とは限らない[5]。

興味傾向の推定における課題は、推定対象である人の興味、個人の主観によるものであるため基準が曖昧で機械的な推定が難しいという点にある。

また、単純に推薦対象者の興味に合致している「利用者/トピック」の推薦を追求していくと、検索によるマッチングになりかねない。そのため興味傾向による推薦はある程度の意外性を含んだものになるべきだと考えられる。

以上より実装する推薦システムは、利用者本人の曖昧な興味を数値化して学習し、利用者本人の興味に合致した推薦が行われ、単純な検索やマッチングでは推薦されない推薦対象者の持つ興味に合致する「利用者/トピック」も推薦される可能性のある「システム/手法」の実現を目指す。

実装するにあたり、利用者の興味を数値として捉えられるのかという問題がある。

よって、前段階として本論文では提案した手法によって利用者の推定が行えるかどうかの検証を目的とする。

具体的には、「いいね」の傾向を基に生成した特徴ベクトルを用いて機械学習を行うことで特定トピックに興味を持つ利用者の傾向推定が可能かの検証を行う。

2. 関連研究

マイクロブログ上で利用者の興味推定を行う既存研究の中に、近藤ら[6]の Twitter 利用者の着目するトピックやフォロー関係を辿り類似する興味を持った利用者の推定を行う手法がある。この手法は各文書におけるトピック分布と各トピックを構成する単語の分布を算出することができる潜在的ディリクレ配分法 (Latent Dirichlet Allocation, LDA) [7]と呼ばれる確率的トピックモデルを用いて、ツイートを対象とした分析を行うものである。具体的には、対象利用者と対象利用者のフォロワーのツイートを収集し、収集したツイートに対して形態素解析を適用する。その後、LDA を適用することにより、各トピックにおける単語の確率分布に対して重みを付け、しきい値を超えたトピックを利用者の興味トピックとして興味推定を行うものである。

この手法は対象利用者の興味推定にフォロワーのツイートを利用しているため、推定結果もフォロワーに依存する。そのため、交友関係が少ない利用者やフォロワーに偏りがある利用者の興味推定には適していない。

また、利用者の投稿からキーワードを抽出する手法として、対象利用者自身の投稿に重み付けを行い、キーワードを抽出するものがある。この手法には興味のある抽出する単語の切り出しの精度や、他の利用者と比較を行う場合に略語や表現方法の違いで同一のトピックを指すキーワードの判別の難しさ、単語に対する重み付けが分散しノイズとなる単語を抽出してしまうといった課題がある。

3. 提案システム

本システム図 1 は、データセット作成処理、興味推定モデル作成処理、投稿群判定処理により構成される。

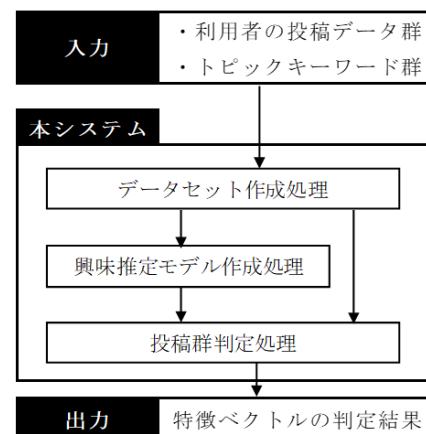


図 1 本システムの処理フロー

†大阪工業大学大学院 情報科学研究科, Graduate School of Information Science and Technology, Osaka Institute of Technology

‡関西大学 環境都市工学部, Faculty of Environmental and Urban Engineering, Kansai University

3.1. 処理フロー

本システムは図 1 より、まず入力データの作成を行い、データセット作成処理、興味推定モデル作成処理、投稿軍判定処理の順で処理が行われ、特徴ベクトルの判定結果が出力で得られるものとなっている。次の節から各処理について解説を行う。

3.2. 入力データ

キーワードの抽出に関する課題の解決には、抽出を行う際に可能な限り興味と関連性の高いサンプルを収集する必要がある。フォロー関係に関する課題の解決には、任意のトピックに対する利用者の興味ベクトルを抽出することで解決する。任意のトピックに興味を抱く人々が、他のトピックに対してどのような興味を抱く傾向があるのかに興味ベクトルとして取得可能であれば、利用者のフォロー関係に依存すること無く興味ベクトルを用いて任意の利用者の興味推定や他の利用者との比較も可能となる。

Twitter 上で主に利用可能な機能として 3 つの機能がある。利用者自身が文章等のコンテンツを投稿する機能である「ツイート」、投稿をフォロワーに拡散する投稿機能である「RT/リツイート」[8](以下、RT)、ツイートに対して好意的な気持ちを示す投稿機能である「いいね/Like」[9] (以下、いいね) である。

先に述べた課題の解決法を実現するにあたり筆者は、Twitter の投稿機能の一部である「いいね」を用いた投稿は利用者の興味と強い相関があると仮説を立てた。

「いいね」を選んだ理由は以下によるものである。

まず「ツイート」は自身の意見や心情等様々なものを投稿する機能であるため興味とは無関係な投稿も多く含まれている。「RT」はフォロワーに対して拡散したい投稿に使用される機能であり自身の投稿に引用する形で使用することも可能である。興味と合致している可能性は高いが、興味とは関係ないが有益で拡散するべきだと感じた投稿も RT されることがある。「いいね」は他人との挨拶等の形式的な意味合いで使用される場合もあるが、基本的に好感や共感した投稿に対して使用される。また、利用者自身は好きだが RT は憚れるような投稿、RT するほどではないが良いと感じた投稿に対しても使用され、後で見るためにブックマークの様な意味合いで使用される場合もある。そのため「いいね」は興味との相関性が強くノイズとなりうる単語が含まれる確率も低いと判断した。

本システムは事前に TwitterAPI[10]を用いて利用者の投稿データ群 (以下、投稿群) と、「いいね」から選出した任意のトピックキーワード群 (以下、単語群) を収集する。この二つのデータを入力として本システムに与える。

3.2.1. 利用者の投稿データ群

任意のトピックキーワードの検索を行い、該当した利用者のユーザーID と取得可能な範囲で利用者の投稿を全てテキストとして収集する。これを投稿群とする。

3.2.2. トピックキーワード群

投稿群の中から無作為に選んだ利用者の「いいね」から、トピックキーワードを抽出し纏める。これを単語群とする。

3.3. データセット作成処理

入力から得た投稿群と単語群を用いて学習用のデータセットを作成する。単語群と利用者の投稿群を比較し、投稿群に単語群中の単語が含まれているなら陽性、それ以外は陰性と判定した特徴ベクトルを利用者毎に作成する。これが利用者の興味傾向を表した興味ベクトルであり、学習時の正解データとなる。また、学習には利用者の投稿を 1 つのテキストに纏め、1 文書の入力として学習を行うため、投稿をテキストから分散表現に変換する必要がある。投稿を分散表現に変換する過程で使用する形態素解析には、形態素解析エンジンである MeCab[11]を利用し、辞書には多数の固有表現が登録されている mecab-ipadic-NEologd[12]を用いる。形態素解析を行った投稿群は、任意の長さの文章を固定長のベクトルに変換する機能を持つDoc2Vec[13]を用いて分散表現に変換する。正解データと分散表現に変換した利用者の投稿がデータセットとなる。

3.4. 興味推定モデル作成処理

3.3 節で作成したデータセットを使用し、入力の次元数を 200、中間の次元数を 120、出力の次元数を特徴ベクトルの次元数とした 8 層の隠れ層を持つ Deep Neural Network (DNN,ディープニューラルネットワーク) にて学習を行い、興味推定モデル (以下、モデル) を生成する。

3.5. 投稿群判定処理

生成されたモデルに任意の利用者の投稿群を与え、モデルの特徴ベクトルの推定結果と正解データを比較する。陰性と判定したものが実際に陰性である場合を真陰性(True Negative, TN)、陰性と判定されたものの、実際には陽性である場合を偽陰性(False Negative, FN)、陽性と判定されたものの、実際には陰性である場合を偽陽性(False Positive, FP)、陽性と判定され、実際に陽性である場合を真陽性(True Positive, TP) と表し、利用者毎にその数を記録し判定結果とする。

3.6. 学習方式

本実験ではニューラルネットワークの階層を深くしたものである。DNN を用いた学習を行う。学習にはニューラルネットワークの構築と訓練を行うため、TensorFlow[14]という機械学習のソフトウェアライブラリを利用する。

4. 事前実験

4.1. 実験内容

特徴ベクトルの生成手法の違いによって発生する差異を確かめる検証実験を行った。

4.2. 単語群の概念分け

3.2.2.項で述べた、単語群を作成する処理について詳しく解説する。

単語群は、利用者の「いいね」から抽出したトピックキーワードの集まりである。単にトピックキーワードを集めただけでは、同じものを指す表現や似たようなものを指す単語を区別できない。そのため、単語群をグループ別に分類し、上位概念と下位概念に振り分ける処理を行った。同じものを指す単語は同一単語として扱った。実際に作成した単語群の抜粋を図 2 に示す。

漫画						
マンガ	週刊少年ジャンプ					
コミック	NARUTO	BORUTO	波の国	疾風伝	鬼滅の刃	鬼滅
	ナルト					

図2 単語群の抜粋

図2では「漫画」という単語が最上位の概念となっており、「BORUTO[*1]」、「波の国」、「疾風伝」が最下位の概念となっている。

上位概念の基準は、細かな概念を一纏めに大別した場合に用いられる概念の単語であるほど上位概念となる。例として「アニメ」、「ゲーム」、「音楽」等が、今回作成した単語群の最上位概念として纏められている。

下位概念の基準は、上位概念に対する概念的な位置づけで決まる。例として、「週刊少年ジャンプ[*2]」は漫画雑誌であり、その雑誌の連載作品であった「NARUTO[*1]」と「鬼滅の刃[*3]」は「漫画」の下位概念となる。更に、「BORUTO」、「波の国」、「疾風伝」は「NARUTO」の作品内に関係する固有名詞であるため「NARUTO」の下位概念となっている。

この様な方式で抽出された単語を振り分けて単語群を作成する。

単語群を作成するにあたり、「いいね」のテキストを形態素解析し単語を機械的に効率よく取り出す試行を試みたが、トピックキーワードとなり得る単語は固有名詞が多く、形態素解析に使用する辞書によっては、「鬼滅の刃」が「鬼/滅/の/刃」として分割された状態でしか抽出されないといった事例が頻発したため、人手による分類を行った。

4.3. 特徴ベクトルの生成

特徴ベクトルの生成時に、単に単語があるか無いかだけを表す one-hot ベクトルを生成するのではなく単語の属するグループ毎への興味傾向を取得するために、特徴ベクトルの生成手法を2通り考え試行した。

単語の探査時に上位概念が検出された場合、その利用者は、下位概念に対しても興味がある可能性が高いと考え、下位概念全てに対して陽性の判定を行うようにした。図3は「NARUTO」が検出された場合の陽性判定の例である。この判定手法を生成手法1とする。

漫画						
マンガ	週刊少年ジャンプ					
コミック	NARUTO	BORUTO	波の国	疾風伝	鬼滅の刃	鬼滅
	ナルト					
0	0	1	0	0	0	0

漫画						
マンガ	週刊少年ジャンプ					
コミック	NARUTO	BORUTO	波の国	疾風伝	鬼滅の刃	鬼滅
	ナルト					
0	0	1	1	1	1	0

図3 NARUTOが検出された場合の判定例

*1: ©岸本斉史 スコット/集英社・テレビ東京・ぴえろ

*2:SHUEISHA Inc. All rights reserved.

*3©吾峠呼世晴/集英社・アニプレックス・ufotable

生成手法2では、生成する特徴ベクトルの配列要素数を増やし、上位概念の検出と下位概念の検出を別々に保持できるようにした。

最上位概念である「漫画」はレベル1、1つ下の概念である「週刊少年ジャンプ」はレベル2といった様に概念毎にレベルを設け、概念に対してそのレベルに応じた要素数を与えた。図4はその例を示したものである。

1	漫画								
2	マンガ	週刊少年ジャンプ							
3	コミック		NARUTO					鬼滅の刃	
4			ナルト	BORUTO	波の国	疾風伝	鬼滅	煉獄	
5							きめつ	煉獄杏寿郎	
6									

↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓

[0] [0, 0] [0, 0, 0] [0, 0, 0, 0] [0, 0, 0, 0] [0, 0, 0, 0] [0, 0, 0] [0, 0, 0, 0]

図4 生成手法2の要素数割り当ての例

レベル2の概念である「週刊少年ジャンプ」は「漫画」に対する判定を記録するために要素数が2つ割り当てられている。レベル3の概念である「NARUTO」、「鬼滅の刃」は、「漫画」「週刊少年ジャンプ」の判定結果を記録するために要素数が3となっている。この方法であれば上位概念の検出を保持したまま下位概念の検出への影響をなくすことが出来る。

例として、投稿全体で「漫画」と「NARUTO」のみが検出されたとする。この場合「漫画」、「週刊少年ジャンプ」、「NARUTO」の配列はそれぞれ[1], [1,0], [1,0,1]という形で記録される。それぞれの要素数の0番目が「漫画」を表し、1番目が「週刊少年ジャンプ」、2番目が「NARUTO」を表す要素となっている。この様に上位概念の検出があったとしても下位概念の検出の有無は影響を受けず保持されることとなる。

4.3.1. 生成手法の検証

生成手法1と生成手法2をそれぞれ採用した本システムに3,240人の利用者のデータを入力として与え、各利用者の特徴ベクトルに陰性判定がどの程度含まれているかを3240人分求め表として纏めた。冒頭の10人を抜粋したものが表1となる。なお、利用者の判別には、Twitterの利用者ユーザーIDナンバーを利用しており、本来はユーザーIDナンバーを表示すべき部分をプライバシー保護の観点から利用者1、利用者2等と表記している。

表1 陰性判定の割合の抜粋

利用者	生成手法1の 陰性判定 の割合(%)	生成手法2の 陰性判定 の割合(%)
利用者1	13	44
利用者2	15	49
利用者3	8	39
利用者4	11	44
利用者5	19	55
利用者6	23	55
利用者7	25	59
利用者8	25	57
利用者9	10	38
利用者10	18	52

表1は冒頭の抜粋だが生成手法1のほうが生成手法2に比べて目に見えて陰性の割合が少ない事が分かる。

3,240人のデータを平均した結果、陰性の割合は34%と低い結果となった。

結果から改めて生成手法1によって作成した正解データとしている特徴ベクトルの配列の値を調べてみると、要素の大半が陽性と判定されていた。

陰性の割合が低くなること自体は想定通りだったものの、3,240人の利用者のデータに対する「漫画」、「アニメ」、「作品」等、最上位概念に対する検出数が想定していたよりも遥かに多い結果となった。最上位概念の検出数の一部の抜粋を表2に示す。

表2 最上位概念の検出総数の抜粋

トピックキーワード	累計検出回数(回)
漫画	2,693
アニメ	2,944
作品	2,394
映画	3,034

入力が3,240人なのに対して最上位概念が検出された回数が2,000を超えているケースが多く、利用者数の半数以上がほとんどの最上位概念に該当する結果となった。

この状態では、各利用者の興味傾向の特徴ベクトルを抽出しようとしているにも拘らず、大半の利用者の特徴ベクトルが、ほとんど同一の物となることが判明した。

生成手法1と同様の手順で生成手法2の検証を行った。

その結果、利用者ごとの陰性判定の割合が3割未満ばかりだった生成手法1と比べて、生成手法2の結果は利用者毎の陰性の割合が6割から4割程度のもまで幅広く出る結果となり、極端な陽性判定への偏りが解消された。

生成手法2の全体のデータを平均した結果、陰性の割合は63%となり生成手法1と比べ29ポイントほど改善された。

最上位概念以外の概念は、上位概念への検出を記録する要素の追加を行ったため、データ全体が若干陰性に偏る傾向にある。

5. 実験

5.1. 実験目的

本実験は、4.3.1.項で検証した結果に基づき、3章で示した本システムの処理を行い、「いいね」の傾向を基に生成した特徴ベクトルを用いた傾向推定の検証を行うことが目的である。

5.2. 実験手順

5.2.1. 実験を行う際に考慮する内容

本実験では「鬼滅の刃」をトピックキーワードとしてデータセットを作成する。「鬼滅の刃」を投稿内容に含む利用者を検索し該当した利用者のユーザーIDと投稿データを収集する。

利用者のデータ収集を行うTwitterAPIの機能には制限[15]があり、15分毎に一定回数以上の情報取得を制限している。そのためアクセス制限を考慮した収集を行う。

5.2.2. 投稿群の作成

今回の実験では収集する利用者を最低3,000人以上、1人の利用者から収集する投稿数は、3000件を基本とし、投稿数が3,000件以下の利用者も収集対象とするが、投稿件数が500件未満の利用者は除外するものとする。本来は投稿件数が3,000件以上の利用者のみを取得し実験を行う計画だったが、TwitterAPIの機能の中でも利用者の検索を行う機能に対する制限[16]が厳しくこの様な基準となった。

5.2.3. 単語群の作成

単語群を生成する過程におけるトピックキーワードの選別は手作業にて選別を行い、654個の単語を抽出し単語群とした。

5.2.4. データセットの作成

生成する特徴ベクトルは4.3.1項の結果から、生成手法2を採用し、1,594次元の特徴ベクトルを作成する。そのため3.4節で述べたDNNの出力次元数は1,594次元となる。

今回の実験ではDoc2Vecにて設定する次元数を200とした。

5.2.5. 興味推定モデルの作成

本実験では汎化性能を評価する統計的な手法である交差検証(cross-validation)を用いて評価する。データをK個に分割してそのうち1つをテストデータに残りのK-1個を学習データとして正解率の評価を行うk分割交差検証法を採用し、分割個数を5個とした。

5.2.6. 投稿群判定処理

学習モデルの推定結果で陽性の確率が0.5以上だった場合を1、それ以外を0として正解データとの比較を行う。

評価に使用するため、最大epoch数に到達した時点での判定結果を出力結果としてcsv形式のファイルに保存する。5分割交差検証法を用いているため、5つの出力結果を平均し、評価を行う。

5.3. 実験内容

実験では、6,940人の利用者を入力として本システムに与え、5.2節の内容に沿って実験を行った。

6. 結果と考察

6.1. 実験の結果と考察

実験を行った結果を示す。3.5節で述べた通り予測結果のパターンをTP, TN, FP, FNで表し、データセットから分割したテストデータの出力結果を累計し、そこから各データの正解率と平均、及び全体の平均を算出し表に纏めたものが表3となる。分割番号は各テストデータを表している。

表3 実験の出力結果の纏め

	正解率(%)	TP(回)	TN(回)	FP(回)	FN(回)	
分割データ番号	1	81.9	541.5540	764.1463	104.0627	184.2370
	2	82.3	506.7853	805.4244	125.6556	156.1347
	3	82.1	466.8919	842.2594	130.4301	154.4186
	4	82.1	449.6902	859.4006	125.3638	159.5454
	5	93.9	499.2233	997.8538	35.1535	61.7695
全体平均	84.5	492.8290	853.8169	104.1331	143.2210	

各テストデータの正解率,陽性の正解率,陰性の正解率をそれぞれ式1,式2,式3によって求め, 5つのデータ全体の平均をとると正解率の平均が 84.5%となった. 陰性の正解率の平均は89.1%, 陽性の正解率の平均は約77.4%と陰性の正解率に比べて低い結果となった.

$$\text{正解率} = \frac{TP+TN}{TP+FP+FN+TN} \quad (1)$$

$$\text{陰性の正解率} = \frac{TN}{FP+TN} \quad (2)$$

$$\text{陽性の正解率} = \frac{TP}{TP+FN} \quad (3)$$

6.2. 考察

実験の結果より, 「いいね」から作成した特徴ベクトルを正解データとして利用者の投稿を推定する学習を行った結果, 8割を超える精度での推定が可能である事がわかった.

よって本研究目的である, 「いいね」の傾向を基に生成した特徴ベクトルを用いて機械学習を行うことで特定トピックに興味を持つ利用者の傾向推定は, 可能であると言える.

実験の結果より陽性の正解率が陰性の正解率よりも低くなった原因は, 特徴ベクトルの作成時に陰性の母数が多くなる生成手法2を採った事により, 陽性に比べて陰性になりやすい要素が多くなった影響だと考えられる. 陰性の偏りを減少させる方法として抽出単語数を増大させ, 下位概念の数が少ない最上位概念を増加させる方法が考えられる. 他には, 単語群の概念振り分けを見直し, 可能な限り下位概念を減らす方法も考えられる. これらの改善策により判定結果の精度向上が可能だと考える.

実験の出力結果のうち, 5番にあたるテストデータの正解率が高い結果となっている. これは投稿群と特徴ベクトルの識別を利用者のユーザーIDで管理し, 学習時のデータの入力を昇順に並べ替えた状態で行った事が原因だと考えられる. ユーザーIDはTwitterアカウントの作成時に割り振られる固有のナンバーであるため, 新規利用者ほど桁数が大きい. つまり昇順に並べた場合, 後半になるほど新規利用者の投稿データとなる. そして収集する利用者の投稿件数は3,000件500件が下限となっている. そのため, 入力データのうち後半のデータとなる5番にあたるテストデータは新規利用者が多く, 投稿件数が少ないグループとなりおり投稿が少ない分検出される単語数が少なくなるため生成される特徴ベクトルに違いが生まれにくく, 推定が他のデータよりも容易だったと推測される.

7. まとめ

本研究では, Twitter利用者の興味傾向を推定するため, 「いいね」を用いた投稿は利用者の興味と強い相関があると仮説を立て, 「いいね」の傾向を基に生成した特徴ベクトルを用いて機械学習を行うことで特定トピックに興味を持つ利用者の傾向推定が可能かの検証を行った. 特徴ベクトルを作成する手法によっては, 生成されたデータに大きな偏りが出る事が確認された. 一方, 「いいね」から作成した特徴ベクトルを正解データとして利用者の投稿を推定する学習を行った結果, 8割を超える精度での予測が可能であり, 「いいね」を用いた傾向推定が可能である事を確認した.

7.1. 改善点

本システムは4.3.1項で判明した様に単語群と特徴ベクトルの作成手法に学習結果が大きく依存するため, 単語の多様性の確保と, 偏りを減少させる必要がある.

また, 6.2節にて書いたように, 入力の方法によってもデータの偏りが生まれていると考えられる. この問題はデータの入力時にデータをランダムに並べ直すことで解決が可能である.

また, 本システムは現在の仕様上, 利用者がどのトピックに対して興味を示す傾向にあるかを推定するものであるため, トピック毎の好悪を判別する事はできない. そのため, 今後追加で実装予定の推薦システムにおいてこの課題を解決する予定である.

推薦システムでは, 本研究で用いたシステムによる出力結果から, 興味傾向が近い利用者を複数人ピックアップし, 投稿に対してネガポジ判定による重みづけや文脈を考慮した判定を行い, 好悪にもマッチする推薦を行う.

7.2. 今後の展望

今後の展望として, まずは学習精度向上のため, 多様な単語選定及び大量獲得の効率化を図る. また, 学習時のパラメータについてもより良い精度が出るよう, 次元数の調整や学習アルゴリズムの選定も併せて行っていく.

次に, 対話可能なTwitter利用者を多数確保し被験者として推定を行い, 人の直感的感性による評価の比較を行う. 利用者間の類似度算出を行い興味が似通っていると判定される利用者として, 人の感覚による興味の傾向にはある程度の乖離があると考えられるため, その点を改善する方法を考える.

また, 現在筆者が参加している地域振興を行うアプリケーション開発プロジェクトにて実証実験も行っていく予定である.

謝辞

本研究に際して着想段階から様々相談に乗って下さり, 課題へのアプローチ方法の思案及びプログラムの検証に快く協力して下さった, 大阪工業大学情報科学部のゼミの同期及び後輩の皆様へ深く感謝いたします.

参考文献

- [1]保高隆之, 報過多時代の人々のメディア選択, ～「情報とメディア利用」世論調査の結果から～, NHK放送文化研究所, <https://www.nhk.or.jp/bunken/research/yoron/20181201_7.html>, 2018.12.01, (入

- 手:2021.02.08).
- [2] Amazon.co.jp, <<https://www.amazon.co.jp/>>, (入手:2021.02.08).
 - [3] Amazon.co.jp, ドライバー検索結果, <<https://www.amazon.co.jp/s?k=%E3%83%89%E3%83%A9%E3%82%A4%E3%83%90%E3%83%BC>>, (入手:2021.02.08).
 - [4] 安田 茂樹,「あなたへのおすすめ」はどう生成するの? Amazon Personalize で簡単に実現する方法をグラレコで解説, AWS 最新ドキュメント紹介, 2020-11-02, (入手:2021.02.08).
 - [5] Twitter, Twitter に表示されるおすすめアカウントについて, <<https://help.twitter.com/ja/using-twitter/account-suggestions>>, (入手:2021.02.08).
 - [6] 近藤 直人, 内田 理: Twitter を用いた LDA に基づく利用者の興味推定手法, 第 21 回年次大会 発表論文集, 言語処理学会, 2015.
 - [7] D. M. Blei, A. Y. Ng, M. I. Jordan, “Latent Dirichlet Allocation”, Journal of Machine Learning Research, Vol. 3, pp.993-1022, 2003.
 - [8] Twitter:リツイートする方法, <<https://help.twitter.com/ja/using-twitter/how-to-retweet>>, (入手:2021.02.08).
 - [9] Twitter:ツイートにいいねする方法, <[help.twitter.com,https://help.twitter.com/ja/using-twitter/liking-tweets-and-moments](https://help.twitter.com/ja/using-twitter/liking-tweets-and-moments)>, (入手:2021.02.08).
 - [10] Twitter:TwitterAPI, <[help.twitter.com,https://developer.twitter.com/ja/docs](https://developer.twitter.com/ja/docs)>, (入手:2021.02.08).
 - [11] MeCab, <<https://taku910.github.io/mecab/>>, (入手:2021.02.08).
 - [12] mecab-ipadic-NEologd:Neologism dictionary for MeCab, <<https://github.com/neologd/mecab-ipadic-neologd>>, (入手:2021.02.08).
 - [13] GENSIM, Doc2vec paragraph embeddings, <<https://radimrehurek.com/gensim/models/doc2vec.html>>, (入手:2021.02.08).
 - [14] TensorFlow, <<https://www.tensorflow.org/learn?hl=ja>>, (入手:2021.02.08).
 - [15] Twitter Developer, Rate limits:Standard v1.1, <<https://developer.twitter.com/en/docs/twitter-api/v1/rate-limits>>, (入手:2021.02.08).
 - [16] Twitter Developer, Search Tweets, <<https://developer.twitter.com/en/docs/twitter-api/v1/tweets/search/guides/standard-operators>>, (入手:2021.02.08).