

# ソーシャルネットワークにおけるフィルターバブルに関する シミュレーションの研究

○水村宏輔 佐々木晃 (法政大学)

## Study of the Simulation on Filter Bubble in Social Networks

\* K. Mizumura and A. Sasaki (University of Hosei)

**概要**— 多くのウェブサイトでは、独自のアルゴリズムにより閲覧履歴などのユーザーのプライベートな情報から推定される、ユーザーに有益な情報の表示を行っている。それにより、同じ意見や思想を持った人たちだけで情報が孤立してしまうことを「フィルターバブル」という。ウェブサイトの運営者は、このような問題を解消しながらアルゴリズムの開発を行なっていかなければならないが、現状、アルゴリズムによる効果をテストできる環境は用意されていない。本研究では、SNS のモデル化を行い、ニュースフィードの表示順を決定するアルゴリズムのマルチエージェントシミュレーションを可能にした。従来利用されてきたアルゴリズムとして、時系列アルゴリズムとエンゲージメント率に基づくアルゴリズムと閲覧履歴に基づくアルゴリズムを作成し、比較シミュレーションを行ない、それぞれの問題点を解決した提案アルゴリズムを作成した。その結果から、ウェブサイトにおいてユーザーにとって有益な情報を見つけやすく、かつ、意図しないフィルターバブルからユーザーを守るためには、閲覧履歴に基づくアルゴリズムで並べられた記事の一部を、ユーザーの嗜好と異なる記事で置き換えればよいと考えられる。

**キーワード:** SNS, フィルターバブル, マルチエージェントシミュレーション, イジングモデル

### 1 まえがき

Twitter や Facebook に代表されるソーシャル・ネットワークキング・サービス (SNS) や、Google, Yahoo! などの検索エンジンなどの多くのウェブサイトでは、独自のアルゴリズムを採用し、閲覧履歴などのユーザーのプライベートな情報からユーザーが必要としていると思われる情報を推定して表示を行なっている。それは、ユーザーにとって大きなメリットである反面、新たな社会問題を生じさせる要因となっている。

その問題の一つとして、インターネット活動家のイーライ・パリサー<sup>1)</sup>が 2011 年に著書の中で提唱した「フィルターバブル」がある。ユーザーはウェブサイトを利用する際に、アルゴリズムが自動的に働くことにより、ウェブサイトの推薦した、ユーザーの嗜好に合っていると思われる情報に囲まれることになる。逆に、ユーザーにとって興味がないと思われる情報は遮断され、インターネット上で見ることができなくなる。これにより、自分と異なった意見の情報から隔離されることから、同じ意見や思想をもった人たちだけで集団が形成され、情報が集団ごとに孤立していくことになる。このような現象をフィルターバブルと呼ぶ。ユーザーがそのようなウェブサイトの特性を理解した上で、自分に必要な情報を効率的に得るために意図的にフィルターバブルを起こしている場合では、そのユーザーに大きな害は生じないと思われる。しかし、誰でも手軽に利用できるウェブサイトでは、多くの人が無意識

のうちにフィルターバブルを作り出してしまっていると思われる。そのようなユーザーは、現在インターネットで見ているものがウェブサイトにより勝手に選ばれたインターネット上のごく一部の情報であるにも拘わらず、世界は自分と同じ意見の人ばかりだと錯覚を起こしてしまう。このような、意図しないフィルターバブルは危険であるといえる。

フィルターバブルを制御するためには、アルゴリズムによる情報の選択を制限することが考えられるが、見たい情報が興味のない情報に埋もれるようなウェブサイトでは、その利用価値は下がってしまうと思われる。本研究では、SNS のニュースフィードに着目したモデル化を行うことにより、アルゴリズムの効果の検証を可能にした。従来の SNS で一般的に使われているようなアルゴリズム(以下、従来アルゴリズムと呼ぶ)を作成し、シミュレーションを行うことにより、ウェブサイトにおいてユーザーにとって有益な情報を見つけやすく、かつ、意図しないフィルターバブルからユーザーを守る方法を提案した。提案アルゴリズムと従来アルゴリズムとの比較シミュレーションの結果から、閲覧履歴に基づくアルゴリズムによって並べられた記事を、ある一定の割合でユーザーと嗜好の異なる記事で置き換える方法により、理想的なアルゴリズムの作成が可能であることが明らかになった。

### 2 関連研究

SNS では、日々ユーザーによって様々な記事が投稿

され、それをまとめて並べられたニュースフィードを閲覧することによってユーザーは情報を得る。SNSの利用者数は年々増加傾向にあり、それにより一定時間あたりに投稿される記事が増加し、ユーザーが自分にとって価値の低い情報に時間を費やしてしまう状況が生じている。そのような状況を改善するために、投稿された多数の記事の中から、ユーザーにとって価値の高い記事を推薦するための研究が多く行われている。

神嵐ら<sup>5)</sup>の研究では、ユーザーの嗜好データを獲得するための方法は、大きく明示的と暗黙的の2つに分けられると述べている。明示的な獲得とは、ユーザーに対して、好き嫌いや関心があるかないかなどの質問をすることにより、嗜好を割り出す方法である。暗黙的な獲得とは、ユーザーの閲覧履歴や投稿履歴、プロフィールなどの個人情報や、記事のエンゲージメント(いいね、リツイート、返信など)などから、嗜好を推定する方法である。データの正確さや未評価と不支持の区別においては明示的な獲得の方が優れているといえる。しかし、SNSにおいてユーザーに適切な記事を推薦するためには、より多くのデータ量が必要であるため、ユーザーに対する質問の量が莫大になってしまうことが考えられる。そのような手間をユーザーは嫌うことが多いため、暗黙的な獲得がSNSにおいて有用であると考えられる。

嗜好データを用いて、ユーザーに適した推薦を行うために、一般的に、ユーザーを嗜好の似通ったグループに分けるクラスタリング手法が利用されている<sup>6)</sup>。桑田ら<sup>4)</sup>の研究では、ユーザーだけでなく、ユーザーに提供されるアイテムに対しても同時にクラスタリングを行う、共クラスタリングという手法を利用して、ユーザーの行動に関する有用なデータを取得しようと試みている。

クラスタリングなどの情報を推薦する手法を利用することによって、SNSの運営側は、他社よりも高い利用満足度を獲得するために、ユーザーと記事を結びつける独自のアルゴリズムを開発している。例えば、Facebook社は、ユーザーに魅力的な情報を評価する方法として、「友人や家族の情報」、「アイデアとしてのプラットフォーム」、「ユーザーが本当に求めている情報か否か」の3点に重点をおいて、ニュースフィードのアルゴリズムの改善や開発を定期的に行い、発表を行っている。また、Twitter社では、見逃している可能性のあるユーザーに対して有益な情報を「最新のハイライト」としてニュースフィードの中に優先的に表示させるなどの工夫を行っている。

SNS上でのユーザーの行動のメカニズムを解明するために、SNSをモデル化することで、ソーシャルメディア上でのユーザー同士の関係性や行動を観察する研究が、数多く行われている。鳥海ら<sup>2)</sup>の研究では、ユーザーの記事投稿や、それに対する返信などの行動の

コストと利得に着目してモデル化を行っている。三ヶ尻<sup>3)</sup>の研究では、ソーシャルメディア上での人間同士に働く同調圧力や情動伝染などの相互作用を踏まえた集団モデルを、2次元イジングモデルで構築している。数多くの研究が行われている中、SNSの運営側が、試作段階のアルゴリズムの効果をテストするためのモデルは考えられていない。したがって、アルゴリズムの効果の検証には実際のSNS上のユーザーを使う方法が考えられるが、それには倫理的な問題が伴う。2012年にFacebook社が行った実験では、実際のFacebookユーザーから無作為に被験者を選び、対象ユーザーのニュースフィードからポジティブやネガティブな言葉を含んだ記事を減らし、ユーザーの記事投稿へどのような影響が生じるかの観察を行なった。結果として、SNSにおいて情動感染が発生することが証明された。しかし、実験が被験者に無断で行われたことから、Facebook社に対する批判が殺到し、謝罪する展開に至った。ユーザーの精神的な面や実社会にまで大きく結びついているSNSにおいて、それに大きな影響を与えるアルゴリズムの開発は、慎重に行わなくてはならない。したがって、アルゴリズムが実際にSNSにおいて利用される前に、モデルを用いてより良いシミュレーションを行える環境を構築することが現在必要とされているのではないかと考えた。本研究においてSNSのニュースフィードに着目したモデル化を行う。

### 3 SNSのモデル化の手法

#### 3.1 SNSのモデル化

本研究では、ユーザーの個人情報と、記事の詳細な情報を利用して、ニュースフィードのアルゴリズムを作成することにより実験を行う。そのため、ユーザーエージェントと記事エージェントを用意し、SNSのモデル化を行った。

ユーザーエージェントの作成は、次の流れで行う

- ・ ユーザーエージェント数の指定
- ・ ユーザーエージェントのクラスタを指定

まず、ユーザーエージェントをN人用意する。SNSでは、それぞれのユーザーが自分の興味のあるユーザーをフォローし、フォローしたユーザーの投稿した記事をニュースフィード上で閲覧することができる。そのような、ユーザー間の複雑な関係をモデルで再現するのは非常に困難である。本研究では、アルゴリズムの違いによるニュースフィードに表示される記事の観察を行うことを目的とするため、ユーザー間の直接的な関係性や個々のユーザーの行動パターンの違いは無視することにする。よって単純化のため、相互にフォローしあっているN人のユーザーエージェントを用意し、学校のクラスのような、お互いにある程度知り合いである閉じた空間のSNSを再現する。

ユーザーはそれぞれ嗜好を持ち、その嗜好に近い記

事を投稿しやすく、また閲覧しやすいといえる。また、嗜好は他のユーザーからの影響を受け変化する。SNSにおいてユーザーの嗜好を推定するためには、一般的に、プロフィール、投稿した記事、クリックした記事などに含まれる情報を集める必要がある。本研究では、記事の内容やプロフィールは考えない。よって、ユーザーの嗜好と記事の内容を結びつける手法として、共クラスタリングが行われていることを前提として研究を進めることにする。具体的には、次の設定を用いる。

- ・ クラスタ数を 100 とする
- ・ ユーザーエージェントは、クラスタ名として 0~99 の値が乱数により与えられる
- ・ 記事エージェントは、投稿したユーザーのクラスタを  $\mu$  とし、 $\sigma$  を 0.5 とする正規分布に従った確率で与えられる
- ・ ユーザーのクラスタは閲覧する記事の影響を受けて変化していく

クラスタ数を 100 とし、ユーザーエージェントと記事エージェントが共クラスタリングされている状態を考える。実際には、ユーザーエージェント作成時に、クラスタ名として、乱数により 0~99 の値が与えられる。尚、クラスタの値には周期境界条件を与える。記事エージェントは投稿するユーザーのクラスタを  $\mu$  とし、 $\sigma$  を 0.5 とする正規分布に従った確率で与えられる。したがって、投稿者に近いクラスタが与えられやすくなる。また、ユーザーのクラスタは閲覧する記事の影響を受けて変化していくように設定する。

ユーザーが SNS を利用する場合、基本的に記事投稿か記事閲覧のいずれかを行うことになる。各ユーザーが SNS を利用するタイミングは各々の私生活に基づき様々である。よって、あるユーザーの一回の SNS 利用を 1 ステップとして、それを繰り返すことによって、ユーザーの記事投稿、または記事閲覧の時間軸における流れを再現する。SNS モデルの流れを Fig. 1 に示す。

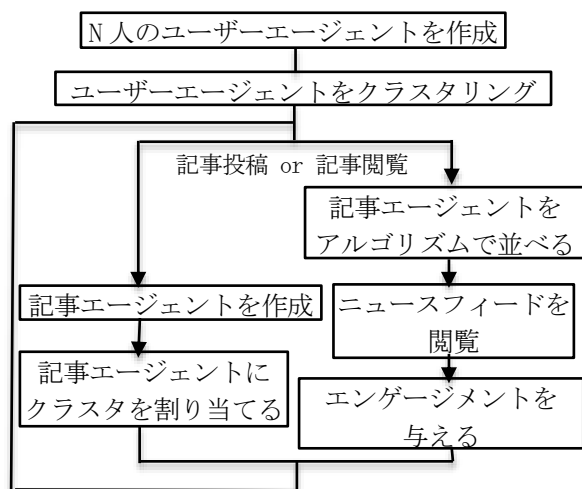


Fig. 1: SNS モデルの流れ.

ステップごとの処理を説明する。まず、N 人の中からランダムで SNS の利用者を 1 人選択する。そのユーザーは、記事投稿、または記事閲覧のいずれかを選択する。記事投稿を選択した場合は、次の流れで処理を行う。

- ・ 記事エージェントを作成
- ・ 記事エージェントのクラスタを指定

まず、ユーザーは記事エージェントを作成する。記事は投稿者、投稿時間、記事のクラスタ、エンゲージメントの情報を保持する。投稿時間には、現在のステップ数が与えられる。記事のクラスタは投稿者のクラスタに近い値が与えられる。実際には、正規分布に従った乱数を生成し、その少数を切り捨てた値が記事エージェントのクラスタになる。記事同士において、クラスタの値の差が小さければ小さいほど、似た内容に関する記事であり、同じ値の記事は同じ内容に関する記事であると表現される。エンゲージメントには、リツイート数、いいね数、閲覧数が含まれ、初期値はそれぞれ 0 である。SNS の運営側はアルゴリズムを作成する際に、これらの記事の情報を利用できる。

記事閲覧を選択した場合は、次の流れで処理を行う。

- ・ 記事エージェントをアルゴリズムで並べる
- ・ 記事に詳細閲覧確率を与える
- ・ 確率に基づいて、ニュースフィードを閲覧
- ・ 記事エージェントに注目するかの判定
- ・ 記事エージェントにリツイートやいいねを与えるかの判定
- ・ 記事エージェントとユーザーエージェントの情報を更新

まず、投稿された記事を、作成したアルゴリズムに従って並べ換える。並べられた記事には詳細閲覧確率が与えられ、それによって記事の詳細を閲覧するか否かを決定する。Facebook の調査によると、ユーザーはニュースフィードを閲覧する際、未読記事の 57% までをスクロールし、それより下の記事を見ようとしない。そのことから、未読記事の 57% を閲覧記事とした。また、未読記事があまりにも多い場合、ユーザーはその 57% も閲覧することができない。本研究では、閲覧記事の上限を 30 記事に定め、それ以降は、閲覧しないように設定した。閲覧記事において、上から順に 1~0 の等差数列による値を与える。さらに、ユーザーは自分の嗜好に合った記事を特に詳細閲覧しやすいことから、ユーザーのクラスタと記事のクラスタの差を計算し、差が 0 の記事を 1 として、差が大きくなるほど、0 に近くなる値を与える。その 2 つの値を掛け合わせた値を詳細閲覧確率とする。記事を詳細閲覧した場合、その記事のクラスタに対してユーザーが注目するか否かを、2 次元イジングモデルを用いて判定する。ユーザーが記事に注目する場合、さらにリツイートやいいねを与えるかの判定を行う。ユーザーエー

エージェントは、詳細閲覧した記事、リツイートをした記事、いいねを与えた記事などの情報を保持する。また、記事エージェントは、閲覧数、リツイート数、いいね数などのエンゲージの情報を更新する。

記事投稿、または記事閲覧が完了すると、次のステップに移りユーザーを再びランダムで選択し、同様の処理を繰り返す。

### 3.2 2次元イジングモデル

イジングモデルは、磁性体の振る舞いを再現したものである。2次元の場合、Fig.2のように格子状にスピンの配置され、それぞれのスピンは、+1か-1のどちらかの値を持ち、周りのスピンの影響を受けて、自らの値を遷移していく。Fig.2において、値が-1である真ん中のスピンの状態遷移を試みる場合、隣接する4つのスピンの値がすべて+1であるので、その影響を受けて、自らも+1に遷移する確率が大きい。反対に、真ん中のスピンの値が+1であった場合、隣接する4つのスピンも同様にすべて+1であるので、-1に遷移する確率は小さくなる。

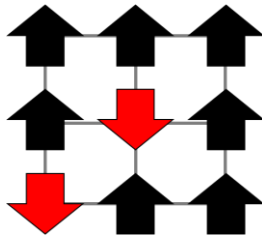


Fig. 2: 2次元イジングモデル。

本研究では、ユーザーがそれぞれの記事のクラスタに対して、現在、注目しているか否かの判定に、そのモデルが使えるのではないかと考えた。まず、記事のクラスタごとに2次元イジングモデルを作成する。尚、スピン間の相互作用については、三ヶ尻<sup>3)</sup>の研究により、Twitterなどのソーシャルネットワークワーキングサービスにおける人間関係を再現するのに適しているパラメータを利用することとする。具体的には次の流れである。

1. エネルギーの関係式(1)とし、エネルギーを表すEの値が小さくなるように各スピンは状態遷移する。人間関係の強度を表すJは、0.047と定義する。

$$E = -\sum_{i=1}^4 J \times S_0 \times S_i \quad (1)$$

2. 状態遷移確率を式(2)で与える。また、温度Tを式(3)で与える。

$$P = \frac{\exp(-\frac{\Delta E}{T})}{2} \quad (2)$$

$$T_i = \frac{1}{1+2 \times |\sum_{j=1}^4 S_{ij}|} \quad (3)$$

格子点をユーザーとし、スピンの値は+1と-1のどちらかで表され、ユーザーが記事に注目しているか否かを表す。初期値は全スピンを-1とする。ユーザーが記事を投稿する際、その記事のクラスタの2次元イジングモデルにおける、そのユーザーのスピンの値を+1にする。ユーザーが記事を詳細閲覧したとき、その記事の2次元イジングモデルにおける、そのユーザーのスピンの値について、状態遷移を試みる。この場合の状態遷移は、ユーザーが記事を見て、その記事に注目するか否かの判定であるので、-1から+1になる状態遷移のみを考える。値がもともと+1であった場合、ユーザーは記事に「いいね」を与える。また、値が-1から+1に遷移した場合、ユーザーは記事を「リツイート」、もしくは近いクラスタの記事を自ら投稿する。また、ステップで選ばれたユーザーに関して、全ての記事クラスタの+1から-1への状態遷移試行を行うことにより、ユーザーの記事に対する注目が、自然に消滅していく現象を再現した。

記事の2次元イジングモデルの情勢Mを(4)で計算し、ユーザー全体からの記事に対する注目度を数値化する。nは格子点の数、Sはあるエージェントjのスピン値を表す。

$$M = \frac{1}{n} \sum_{j=1}^n S_j \quad (4)$$

Mの値は0に近い状態からスタートし、その状態が続いている場合、SNSにおいて、その記事の情勢は安定しているといえる。また、あるきっかけを境目に、Mの値が1に近い状態に落ち着いてしまうことを相転移といい、その状況になっている記事は、多くのユーザーから大きな注目を集めている記事であり、いわば「炎上」状態にあるといえる。炎上とは、一般的に悪い意味で情報に注目が集まることにより、投稿者や関連する者に批判が殺到することを意味するが、本研究においては、称賛を集めた記事に対する注目度の上昇も含めて炎上と呼ぶことにする。

### 3.3 従来アルゴリズム

従来アルゴリズムとして、SNSの運営側が獲得できる情報を利用した、比較的単純であり、運用した際の結果が明らかであるニュースフィードのアルゴリズムを作成し、モデルにより比較シミュレーションを行う。本研究では、時系列アルゴリズム、エンゲージメント率に基づくアルゴリズム、閲覧履歴に基づくアルゴリズムの3つを作成した。

SNSのニュースフィードに記事を並べる方法として最もシンプルなもの、投稿された記事を、時系列に並べていく方法である。これによりユーザーは、リアルタイムで「現在」投稿されている情報を優先的に見ることができる。この方法では、ユーザーの情報や記事の情報による記事の選択が行われないので、フィルターバブルが起こることは考えられない。しかし、記事が頻繁に投稿されるSNSでは、ユーザーにとって価値のある記事が投稿された後に、価値のない記事が多数投稿されることによって、記事を見逃してしまったり、SNSに費やす無駄な時間が増えてしまったりする。このようなSNSでは、ユーザーが不便を感じてしまうことが考えられる。この方法は、何もアルゴリズムが働いていない状態であるといえるが、本論文では、時系列アルゴリズム(Fig. 3)と呼ぶことにする。

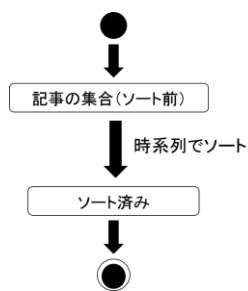


Fig. 3: 時系列アルゴリズム.

記事の情報から、ニュースフィードの並び方を考える場合に、一般的に利用されるのはエンゲージメント率である。エンゲージメント率とは、一般的に記事に与えられるリツイート数、いいね数などのエンゲージメントを閲覧数で割った値である。エンゲージメント率が高い記事ほど、SNSにおいてユーザーから多く注目されている記事であるといえる。エンゲージメントの算出方法はSNSによって独自に決められている。本研究では、リツイートの方が、いいねより重要であると考え、リツイート数に5をかけた値といいね数に1をかけた値の和をエンゲージメントとし、(5)により、エンゲージメント率を求めることにする。

$$E = \frac{(\text{retweet} \times 5) + (\text{favorite} \times 1)}{\text{look}} \quad (5)$$

エンゲージメント率の高い記事を、ユーザーのニュースフィードに優先的に表示させるアルゴリズムを考える。実装の方法として、投稿された記事に時系列順にスコアを与え、それに、エンゲージメント率の大きさに応じた値を足し合わせて、スコア順に並べ替える。この方法では、ユーザーの個々の嗜好は考慮されず、全てのユーザーにエンゲージメント率の高い記事が表示される。それにより、注目を集めた記事が全てのユーザーの目に入りやすくなり、さらに注目を集めるこ

とにより多く拡散され、全てのユーザーのニュースフィードに表示される記事がその記事に関するものばかりに偏ってしまうと考えられる。この方法を、エンゲージメント率に基づくアルゴリズム(Fig. 4)と呼ぶことにする。

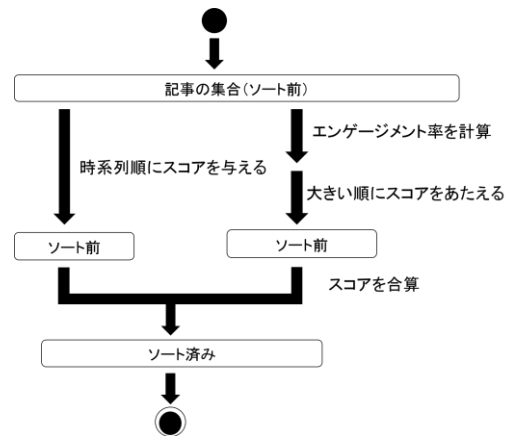


Fig. 4: エンゲージメント率に基づくアルゴリズム.

ユーザー個々の情報から、ニュースフィードを並び替える方法を考える。一般的に考えられる方法として、ユーザーの過去の行動から、ユーザーの嗜好を推定し、その嗜好に合った記事を表示する方法がある。実装の方法として、投稿された記事に時系列順にスコアを与え、ユーザーが過去に詳細閲覧、いいね、リツイートを行った記事のクラスタと同じクラスタの記事にスコアを加算し、再びスコア順に並べ替える。この方法では、ユーザーのニュースフィードは、現在、多くの人に注目されている記事の影響を受けず、それぞれ自分の嗜好に近い記事が表示されやすくなる。それにより、嗜好の近いユーザー同士でニュースフィードが形成され、その他の嗜好の人の記事が表示されにくくなる。よって、ユーザーのニュースフィードは自分のクラスタに近い記事で構成され、自分と違うクラスタの記事はほとんど見えなくなると考えられる。この方法を、閲覧履歴に基づくアルゴリズムと呼ぶことにする。

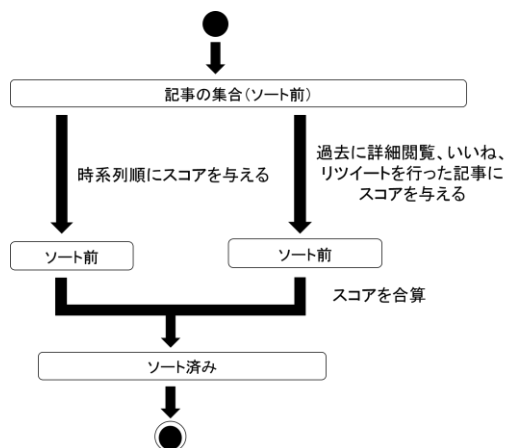


Fig. 5: 閲覧履歴に基づくアルゴリズム.

### 3.4 シミュレーション結果

作成した SNS モデルの有用性を検証するために、予備実験として従来アルゴリズムによるシミュレーションを行った。ユーザーエージェントの数を  $35 \times 35$  の 1225 人、ステップ数 4000 回とした。また、実際の SNS を再現するために、十分な量の記事が投稿された状態で実験を開始する必要がある。そのために、時系列アルゴリズムを 5000 回、測定するアルゴリズムを 5000 回、計 10000 回の空回しを行った後から、計測を開始することにする。

従来アルゴリズムにより、記事クラスタごとの情勢  $M$  の推移の比較シミュレーションを行った。投稿された記事の注目度を Fig 6, Fig 7, Fig 8 に示す。

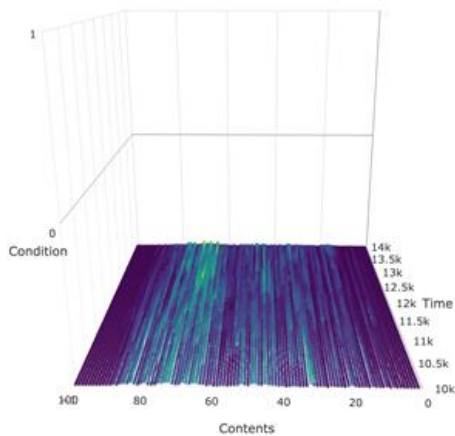


Fig. 6: 時系列アルゴリズムにおける情勢.

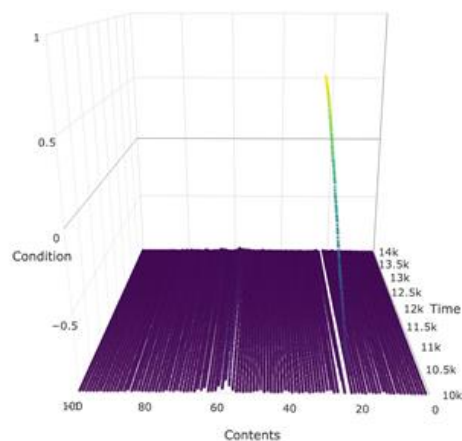


Fig 7: エンゲージメント率に基づくアルゴリズムにおける情勢.

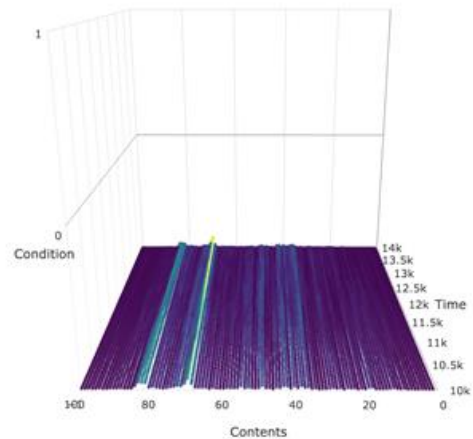


Fig 8: 閲覧履歴に基づくアルゴリズムにおける情勢

横軸が記事クラスタを表し、縦軸が記事の情勢  $M$  を表し、奥行きは時間軸である。記事の情勢は、どのくらいのユーザーが記事クラスタに注目しているかを表す。その値が 0 を超えた場合、半数以上が記事クラスタに注目している状態を表し、炎上状態にあるといえる。3つのアルゴリズムの比較を行う。時系列アルゴリズムでは、注目度の上昇は起こっておらず、常に安定した状態が続いている。エンゲージメント率に基づくアルゴリズムでは、1つの記事クラスタにおいて大きく注目度が上昇し、炎上している状態に至った。それ以外の記事クラスタでは、あまり注目度の上昇が起こらないという結果となった。閲覧履歴に基づくアルゴリズムでは、大きな注目度の上昇は確認できず、小さな山が複数確認できる程度にとどまった。この結果から、エンゲージメント率に基づくアルゴリズムにおいて、ユーザーの注目が1つの記事に集中し、炎上が起こることがわかった。また、時系列アルゴリズムと閲覧履歴に基づくアルゴリズムでは、大きな注目の上昇は起こりづらい。閲覧履歴に基づくアルゴリズムでは、複数の記事クラスタにおいて、小さな注目度の山が実験の開始から終了まで続いている。このことから、閲覧履歴に基づくアルゴリズムでは、ユーザーの全体的な注目度の上昇は起きないが、複数の小さなユーザーの集まりが形成され、その中でひとつのクラスタの記事が注目を集め続けることがわかった。これらの結果からこのモデルを使ってアルゴリズムによる影響を観察することは可能であることが明らかになった。

さらに、3つのアルゴリズムにより、ユーザーが記事を読覧する際、ニュースフィードに表示される記事のクラスタの分布の比較シミュレーションを行った。ニュースフィードに表示される上位 30 個の記事のクラスタの相対度数分布を、ヒートマップにまとめ、Fig 9, Fig 10, Fig 11 に示す。

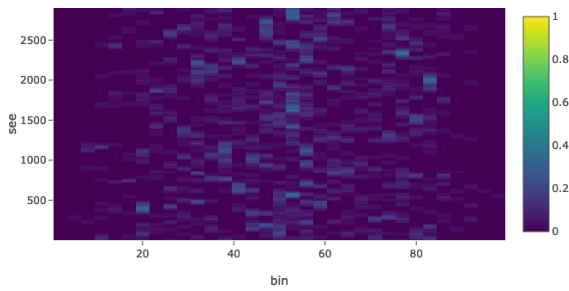


Fig 9: 時系列アルゴリズムにおける相対度数分布.

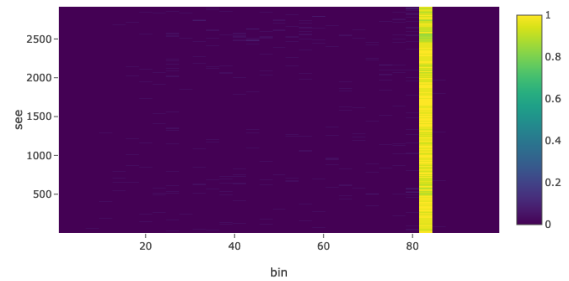


Fig 10: エンゲージメント率に基づくアルゴリズムにおける相対度数分布.

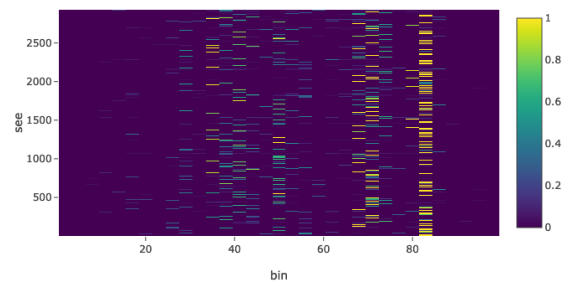


Fig 11: 閲覧履歴に基づくアルゴリズムにおける相対度数分布.

横軸が記事クラスタを表し、縦軸がステップの順番を表し、色の違いによりニュースフィードに表示されている記事クラスタの分布を表している。時系列アルゴリズムでは、比較的記事のクラスタの偏りは少なく、相対度数が 0.4 を超える値は確認できなかった。これは、時系列アルゴリズムでは、フィルターバブルが起

こり得ないという仮説を証明する結果となった。エンゲージメント率に基づくアルゴリズムでは、閲覧したほとんどユーザーにおいて表示されるクラスタが共通して一つの値に偏っている。これは、エンゲージメント率に基づくアルゴリズムでは、全ユーザーのニュースフィードが 1つの記事に関する記事ばかりに偏ってしまうという仮説を証明する結果となった。閲覧履歴に基づくアルゴリズムでは、様々なクラスタにおいて点々と偏りが起こっており、閲覧するユーザーによって、異なったクラスタでニュースフィードが偏っているといえる。これは、閲覧履歴に基づくアルゴリズムでは、ユーザーごとに自分のクラスタに近い記事ばかりが表示されることにより、それ以外の記事は表示されにくくなるという仮説を証明する結果となった。したがって、それぞれのアルゴリズムにおいて、その特徴に沿った結果となり、この SNS モデルの有用性が明らかになった。

## 4 評価法

### 4.1 従来アルゴリズムの問題点

予備実験の結果から、3つの従来アルゴリズムにおける問題点をまとめる。

時系列アルゴリズムにおいて、記事クラスタの情勢は安定しており、また、ユーザーが記事を閲覧した際のニュースフィードに表示される記事のクラスタにも大きな偏りは現れなかった。このことから、このアルゴリズムではフィルターバブルが起こらないことは明らかになった。しかし、頻繁に記事が投稿される SNS において、見たい情報が興味のない情報に埋もれてしまい、ユーザーが不便を感じてしまうという、根本の問題の解決が必要である。

エンゲージメント率に基づくアルゴリズムにおいて、記事クラスタの情勢  $M$  は、ある 1つのクラスタにおいて相転移を起こしている。それは、ユーザーの半数以上がその記事に注目している状態であり、実際の SNS では「炎上」を意味する状態であるといえる。ユーザーが記事を閲覧した際のニュースフィードに表示される記事のクラスタの状況を見ても、ほとんどのユーザーにおいて、同じクラスタの記事がニュースフィードを埋め尽くす結果となった。これはフィルターバブルが起きていることを意味する。全てのユーザーは自分の嗜好とは異なっても SNS 上でひとつのクラスタの記事ばかりを見ることになり、そのクラスタに注目せざるを得ない状況に陥る。それにより、SNS によって同調圧力が引き起こされる可能性があり、危険な状態であると考えられる。

閲覧履歴に基づくアルゴリズムにおいて、ユーザーが記事を閲覧した際のニュースフィードに表示される記事のクラスタは、閲覧したユーザーごとに異なったクラスタにおいて偏りが起きることがわかった。ユー

ザーはそれぞれ自分の嗜好に近い記事を読覧、またはエンゲージを与える。それらの行動の履歴から、アルゴリズムが記事を推薦することによって、同じクラスタの記事が表示されやすくなる。それにより、クラスタの近いユーザー同士でフィルターバブルを起こし、他のクラスタの記事はあまり表示されないという状況が起きる。ユーザーは自分の意見に合った記事ばかり目にするので、興味のない記事に時間を費やすことはないが、自分と異なる意見の存在が見えなくなってしまうので、SNS を利用しているユーザー全体が自分と同じ意見であると勘違いしてしまう可能性がある。

これらのことから、提案アルゴリズムを作成する際に、フィルターバブルが起こっているか否かと、ニュースフィードにユーザーにとって興味のない記事が混ざっているかの2つの点で評価する必要があると考えられる。

#### 4.2 フィルターバブルの評価

フィルターバブルは、ユーザーがニュースフィードにおいて偏った情報しか得られなくなることを意味する。本研究では、記事は0～99のクラスタによって分けられ、その値によって内容の近い記事と遠い記事の判別ができる。よって、ニュースフィードに表示される記事クラスタにどれだけ偏りがあるかを数値化することにより、フィルターバブルが起こっているか否かの判定ができると考えられる。偏りを数値化する方法として、標準偏差を利用する。ユーザーが記事を読覧したとき、ニュースフィードに表示される上位30個の記事クラスタについて、標準偏差を計算する。それにより、読覧したユーザーにおけるニュースフィードの記事の散らばりを数値化できる。シミュレーション中にユーザーが行なった記事読覧全てに対して標準偏差を計算し、その値の平均を出すことにより、実験に使ったアルゴリズムにおけるフィルターバブルの評価を行う。求められた値をニュースフィードの標準偏差と呼ぶことにする。

#### 4.3 ニュースフィードに含まれる興味のない記事における評価

ニュースフィードに表示される記事にフィルターバブルが起こっていないとしても、自分にとって興味のない記事が多く表示され、見たい記事を探すのに苦労しなければいけない場合、ユーザーは不便を感じるだろう。本研究では、記事は0～99のクラスタによって分けられ、ユーザーも自らの嗜好を表す0～99のクラスタによって分けられる。記事クラスタとユーザークラスタは相互に対応している。つまり、ユーザーのクラスタが10であった場合、そのユーザーにとって見たい情報はクラスタ10の記事、もしくはそれに近い9や11の記事であり、60などの値の遠く離れ

た記事は、そのユーザーにとって興味のない記事であるといえる。つまり、ユーザーが記事読覧をしたときにニュースフィードに表示される記事のクラスタがユーザークラスタと離れている場合、興味のない記事が表示されていることになる。

ユーザーにとって興味のない記事がニュースフィードにどのくらい表示されているかを数値化する方法を考える。ユーザーが記事を読覧したとき、ニュースフィードに表示される上位30個の記事のクラスタについて、ユーザークラスタとの値の差の絶対値を計算し、その30個の値の平均値をとる。それにより、読覧したユーザーのニュースフィードがどのくらい不便であるかを数値化できる。その値を不便指数と呼ぶことにする。シミュレーション中に起きた記事読覧全てに対して不便指数を計算し、その値の平均を出すことにより、実験に使ったアルゴリズムにおける不便指数の評価を行う。

#### 4.4 従来アルゴリズムの評価実験

3つの従来アルゴリズムにおいて、再度シミュレーションを行い、ニュースフィードの標準偏差と不便指数を計算する。条件は予備実験と同じく、ユーザーエージェントの数を1225人、ステップ数4000回とした。実験はそれぞれのアルゴリズムにおいて10回ずつ行い、結果の平均をTable 1に示す。

Table 1: 従来アルゴリズムの評価実験結果。

	TIME	ENGAGE	HISTORY
標準偏差	18.95	3.94	7.01
不便指数	19.50	16.03	12.51

3つのアルゴリズムの結果を比較すると、まずニュースフィードの標準偏差において、時系列アルゴリズムでは18.95、エンゲージメント率に基づくアルゴリズムでは3.94、閲覧履歴に基づくアルゴリズムでは7.01と、時系列アルゴリズムと他の2つのアルゴリズムとの間に大きな違いが表れた。このことから、エンゲージメント率に基づくアルゴリズムと、閲覧履歴に基づくアルゴリズムにおいてニュースフィードに表示される記事に偏りがあることが数値によって明らかになった。したがって、フィルターバブルの数値化がニュースフィードの標準偏差によって可能であることがわかった。

不便指数において、時系列アルゴリズムでは19.50、エンゲージメント率に基づくアルゴリズムでは16.03、閲覧履歴に基づくアルゴリズムでは12.51となった。時系列アルゴリズムでは、ユーザーに記事の推薦を行わず、投稿された順に記事が表示されていくので、必然的に表示される記事には見たい記事と興味のない記事が混ざった状態で表示される。それに比較して、エ



ンゲージメント率に基づくアルゴリズムでは、ユーザーが注目している記事が全ユーザーのニュースフィードに表示されやすくなるので、その記事と嗜好が近いユーザーにとって、不便指数は小さくなり、嗜好が遠いユーザーにとっては不便指数が大きくなる。しかし、ユーザーは目に入る記事に影響を受けて嗜好が変化していくので、ユーザーの嗜好自体が一点に集まり、全体の不便指数が下がっていくという結果になったといえる。閲覧履歴に基づくアルゴリズムでは、ユーザーごとに自分の嗜好の近い記事がニュースフィードに表示されやすくなる。それにより、最も不便指数が下がる結果になったといえる。したがって、ユーザーの記事閲覧における不便さの数値化が不便指数によって可能であることがわかった。

## 5 フィルターバブルに関するマルチエージェントシミュレーション

### 5.1 提案アルゴリズム

ユーザーが記事を読覧するとき、ニュースフィードに表示される記事に偏りが無い点と、見たい記事が興味のない記事に埋もれていない点に着目して提案アルゴリズムの作成を行う。つまり、本研究の評価法において、ニュースフィードの標準偏差がより大きく、かつ、不便指数がより小さくなるようなアルゴリズムを目的とする。

従来アルゴリズムの評価実験の結果から、ニュースフィードの標準偏差が最も大きいアルゴリズムは、時系列アルゴリズムの 18.95 であり、最も小さいアルゴリズムはエンゲージメント率に基づくアルゴリズムの 3.94 である。また、不便指数が最も小さいアルゴリズムは閲覧履歴に基づくアルゴリズムの 12.51 で、最も大きいアルゴリズムは時系列に基づくアルゴリズムの 19.50 である。よって、提案アルゴリズムは、ニュースフィードの標準偏差において、時系列アルゴリズムの値に近づき、かつ、不便指数において、閲覧履歴に基づくアルゴリズムの値に近くなることが理想であると考えられる。

提案アルゴリズムの作成方法を考える。閲覧履歴に基づくアルゴリズムにより、ユーザーが記事閲覧をしたとき、ニュースフィードに自分の嗜好にあった記事が表示されやすくなるという結果から、このアルゴリズムをベースにして、自分と嗜好の異なる記事の存在にも気づけるようなアルゴリズムを考える。その方法として、閲覧履歴に基づくアルゴリズムによって並べられた記事を、ある割合でユーザーの嗜好と異なる記事で置き換えるということが考えられる。ユーザーの嗜好と異なる記事の中でより注目されている記事を、そのユーザーのニュースフィードに入れることによって、ユーザーは自分と違う意見の人が多数存在することを知ることができる。しかし、嗜好の異なる記事で

置き換える割合を大きくしすぎると、ニュースフィードが自分の嗜好の違う記事ばかりになり、不便指数が上がってしまうことが考えられる。また、割合を小さくしすぎると、ニュースフィードは自分と嗜好の合った記事で偏っているままである。嗜好の異なる記事で置き換える割合のパラメーターを動かすことによって、どのような結果の違いが生まれるか、シミュレーションを行い検証する。

### 5.2 提案アルゴリズムのシミュレーション

提案アルゴリズムにおいてシミュレーションを行い、ニュースフィードの標準偏差と不便指数を計算する。閲覧履歴に基づくアルゴリズムによって並べられた記事を、嗜好の異なる記事で置き換える割合による結果の違いを比較する。挿入する割合を 0%~60%まで 3%ずつ上げていき、ニュースフィードの標準偏差と不便指数の推移を調べる。従来アルゴリズムによる結果と比較を行うため、条件は同様にして、ユーザーエージェントの数を 1225 人、ステップ数 4000 回とした。実験はそれぞれ 10 回ずつ行い、その結果の平均を Fig 13, Fig 14 にまとめた。

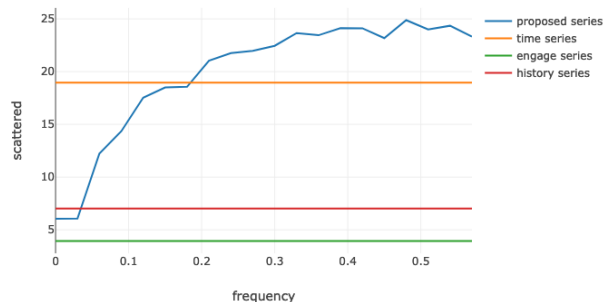


Fig 13: 提案アルゴリズムにおけるニュースフィードの標準偏差。

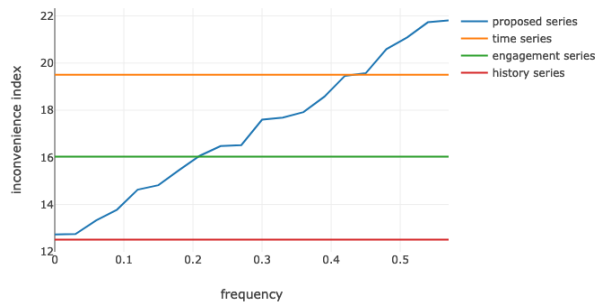


Fig 14: 提案アルゴリズムにおける不便指数。

青色の線が提案アルゴリズム、オレンジ色の線が時系列アルゴリズム、緑色がエンゲージメント率に基づくアルゴリズム、赤色が閲覧履歴に基づくアルゴリズムを示している。

Fig 13 を見ると、ニュースフィードの標準偏差において、嗜好の異なる記事で置き換える割合が 0%~20% の範囲において値が大きく上昇していき、それ以降では、比較的値は定常状態になるという結果となった。従来アルゴリズムの中で最もニュースフィードの標準偏差の大きかった時系列アルゴリズムより値が小さく、その次に大きかった閲覧履歴に基づくアルゴリズムより値が大きくなる範囲に注目する。提案アルゴリズムにおいて、嗜好の異なる記事で 6%置き換えたときの値は 12.24 で、時系列アルゴリズムの 18.95 との差が 6.71、また、閲覧履歴に基づくアルゴリズムの 7.01 との差は 5.23 であった。嗜好の異なる記事で置き換える割合が 6%より小さい場合では、いずれの割合においても、ニュースフィードの標準偏差の値は 12.24 よりも小さい値となっている。よって、6%以下を嗜好の異なる記事で置き換えることにより、ニュースフィードの標準偏差は時系列アルゴリズムよりも閲覧履歴に基づくアルゴリズムに近い値となることがわかった。嗜好の異なる記事で 9%置き換えたときの値は 14.35 で、時系列アルゴリズムとの差が 4.60、また、閲覧履歴に基づくアルゴリズムとの差は 7.34 であった。嗜好の異なる記事で置き換える割合が 9%より大きい場合では、いずれの割合においても、ニュースフィードの標準偏差は 14.35 よりも大きい値となっている。よって、9%以上を嗜好の異なる記事で置き換えることにより、ニュースフィードの標準偏差は時系列アルゴリズムに最も近い値になることがわかった。

Fig 14 を見ると、不便指数は一次関数的に一定の傾きで上昇していく結果となった。従来アルゴリズムと比較して、最も不便指数の小さかった閲覧履歴に基づくアルゴリズムより値が大きく、その次に小さかったエンゲージメント率に基づくアルゴリズムより値が小さくなる範囲に注目する。提案アルゴリズムにおいて、嗜好の異なる記事で 9%置き換えたときの値は 13.78 で、閲覧履歴に基づくアルゴリズムの 12.51 との差が 1.27、また、エンゲージメント率に基づくアルゴリズムの 16.03 との差は 2.25 であった。嗜好の異なる記事で置き換える割合が 9%より小さい場合では、いずれの割合においても、不便指数の値は 13.78 よりも小さい値となっている。よって、9%以下を嗜好の異なる記事で置き換えることにより、不便指数は閲覧履歴に基づくアルゴリズムに最も近い値になることが分かった。嗜好の異なる記事で 12%置き換えたときの値は 14.63 で、閲覧履歴に基づくアルゴリズムとの差が 2.12、エンゲージメント率に基づくアルゴリズムとの差が 1.40 であった。嗜好の異なる記事で置き換える割合が 12%

より大きい場合では、いずれの割合においても、不便指数は 14.63 よりも大きい値となっている。よって、12%以上を嗜好の異なる記事で置き換えることにより、不便指数は閲覧履歴に基づくアルゴリズムよりもエンゲージメント率に基づくアルゴリズムに近い値となることがわかった。

### 5.3 考察

提案アルゴリズムを作成する際に目標として、ニュースフィードの標準偏差において、時系列アルゴリズムの値に近づき、かつ、不便指数において、閲覧履歴に基づくアルゴリズムの値に近くなることと定めた。シミュレーションの結果から、この目標を達成できる条件について考える。

まず、ニュースフィードの標準偏差について考える。シミュレーションの結果から、嗜好の異なる記事を挿入する割合が 9%以上の場合において、ニュースフィードの標準偏差はエンゲージメント率に基づくアルゴリズムと閲覧履歴に基づくアルゴリズムよりも、時系列アルゴリズムに近い値となることが明らかとなった。よって、提案アルゴリズム作成の目標の 1 つである、標準偏差が時系列アルゴリズムの値に近づくという点で、閲覧履歴に基づくアルゴリズムの 9%以上を嗜好の異なる記事で置き換えることによって達成できることがわかった。

次に、不便指数について考える。実験の結果から、嗜好の異なる記事を挿入する割合が 9%以下の場合において、不便指数は時系列アルゴリズムとエンゲージメント率に基づくアルゴリズムよりも閲覧履歴に基づくアルゴリズムに近い値となることが明らかとなった。よって、提案アルゴリズム作成の目標のもう 1 つである、不便指数が閲覧履歴に基づくという点で、閲覧履歴に基づくアルゴリズムの 9%以下を嗜好の異なる記事で置き換えることによって場合において達成できることがわかった。

したがって、閲覧履歴に基づくアルゴリズムにより並べた記事の 9%をユーザーの嗜好と異なる記事で置き換えることによって、ニュースフィードの標準偏差において、時系列アルゴリズムの値に近づき、かつ、不便指数において、閲覧履歴に基づくアルゴリズムの値に近くなるアルゴリズムを作成することが可能であることが明らかになった。ただし、9%置き換えた場合では、ニュースフィードの標準偏差において、理想値からは 4.60 下回り、不便指数において、理想値からは 1.27 上回る。ニュースフィードの標準偏差をさらに理想に近づけるためには、置き換える割合を 9%より大きくする必要がある。ニュースフィードの標準偏差は 0%~20%の範囲で急激に上昇していくので、パラメーターを少し上げるだけで十分な効果が出るといえる。不便指数においては、実験を行った 0%~60%の範囲では一次関数的に上昇していくので、パラメーターを上

げた分に比例して、その値も上昇していく。よって、置き換える割合を大きくすればするだけ、不便指数は理想値から遠ざかっていく。不便指数をさらに理想に近づけるためには、置き換える割合を9%より小さくする必要がある。不便指数においては、小さくすればするだけ理想値に近づいていくが、ニュースフィードの標準偏差はパラメーターを少し下げただけでその値は大きく減少してしまう。そのことから、パラメーターを下げる際には注意が必要である。

これらのことから、本研究のモデルにおいて、ユーザーエージェントの数を1225人、ステップ数4000回とした場合において、閲覧履歴に基づくアルゴリズムを嗜好の異なる記事で置き換える割合が9%であるときに、提案アルゴリズムの理想を達成することが明らかになった。ただし、実際のSNSにおいて、本研究におけるユーザー数やステップ数のパラメーターは常に変化していく値であるので、今回のシミュレーションの結果が実際のSNSに適応することは明らかではない。実際のSNSにおいて、提案アルゴリズムの効果を実証するためには、さらに、実データとの比較を行うことや、ユーザー数やステップ数により、結果に変化が生じるか否かをさらに検証する必要がある。

本研究の結果を、実際のSNSに置き換えて考えてみると、ニュースフィードの標準偏差の結果から、従来の閲覧履歴に基づくアルゴリズムにより、ニュースフィードの記事に生じていた偏りが、ユーザーの嗜好と異なる記事が全体の9%でも表示されていれば解消するということになる。つまり、ユーザーのニュースフィードに並んでいる記事のほとんどがユーザーと意見や思想の近い内容の記事であったとしても、異なる意見や思想の記事をわずかに表示させるだけで、フィルターバブルの危険性は大きく減少するということになる。ニュースフィードにおいて、ユーザーと全く嗜好の異なる記事を表示することによって、ユーザーにその存在を大きなインパクトと共に伝えることができるので、時系列のような嗜好を考慮して推薦を行っていないニュースフィードのように、ユーザーに多様な意見や思想が存在することを伝えることができると考えられる。

また、不便指数の結果から、ユーザーと嗜好の違う記事が全体の9%より多く表示されている場合において、ユーザーは不便を感じるようになる。ユーザーはそもそもSNSにおいて自分の興味のある記事をチェックすることを目的としている場合が多い。よって、自分と嗜好の異なる記事がニュースフィードに多く存在すればするほど、ユーザーは興味のある記事を見るために、より多くの時間を費やさなければならない。9%以下を置き換えた場合においては、ユーザーは自分の興味のある記事を開覧するのに苦労することはあまりないといえる。しかし、ユーザーと嗜好の異なる記

事として、ユーザーにとって不快な記事や、ユーザーに悪影響を与える記事が表示された場合、ユーザーはそのSNSの利用を控えてしまう可能性がある。その点を解決するためには、推薦システムなどにより、そのような記事を分類しユーザーのニュースフィードに表示させないように、さらにフィルターをかける必要があると考えられる。

## 6 むすび

多くのウェブサイトでは、独自のアルゴリズムにより、閲覧履歴などのユーザーのプライベートな情報から推定される、ユーザーに有益な情報を推定し表示を行っている。それにより、同じ意見や思想を持った人たちだけで情報が孤立してしまう「フィルターバブル」という現象が問題になっている。自分がフィルターバブルの中にいることを認識できていない状態は危険であると言える。ユーザーの意図しないフィルターバブルを制御するためには、アルゴリズムによる情報の選択を制限することが考えられるが、見たい情報が興味のない情報に埋もれるようなウェブサイトでは、その利用価値は下がってしまうと思われる。ウェブサイトの運営側は、その利用価値を下げることなく、意図しないフィルターバブルなどのユーザーにとって危険な状態を回避するようなアルゴリズムの開発を行わなければならない。従来の研究では、ユーザーに適したトピックの推薦を行う手法の検討は行われているものの、SNSの運営側が、その手法により開発したアルゴリズムの効果をテストするためのモデルは考えられていない。アルゴリズムのテストを実際のユーザーを使って行うには、倫理的な問題が伴う。よって、本研究では、SNSにおけるニュースフィードのアルゴリズムに着目したモデル化を行ない、アルゴリズムが実際に使用されるまでのテスト環境を構築した。そのモデルを使って、マルチエージェントシミュレーションを行うことにより、ウェブサイトにおいてユーザーにとって有益な情報を見つけやすく、かつ、意図しないフィルターバブルからユーザーを守る方法の提案を行った。

従来のSNSのニュースフィードのアルゴリズムとして、記事を投稿された順番に並べた時系列アルゴリズムと、記事に与えられたエンゲージメントにより並び替えられる、エンゲージメント率に基づくアルゴリズムと、ユーザーの閲覧履歴に従って並べ替えられる、閲覧履歴に基づくアルゴリズムの、計3つのアルゴリズムで比較シミュレーションを行なった。結果として、エンゲージメント率に基づくアルゴリズムと閲覧履歴に基づくアルゴリズムにおいて、表示される記事に偏りが表れた。また、時系列アルゴリズムでは、記事の偏りは表れなかったが、ユーザーの嗜好に合った記事の推薦が行われていない。この結果から、アルゴリズムの評価を行うために、ニュースフィードに表示され

る記事に散らばりがどれくらいあるかを表すニュースフィードの標準偏差と、ユーザーの嗜好と異なった記事がどのくらい表示されているかを表す不便指数を計算し、比較を行なった。結果として、ニュースフィードの標準偏差において、時系列アルゴリズムが最も小さい値となり、不便指数においては、閲覧履歴に基づくアルゴリズムにおいて、最も小さい値となった。よって、提案アルゴリズムは、ニュースフィードの標準偏差において、時系列アルゴリズムの値に近づき、かつ、不便指数において、閲覧履歴に基づくアルゴリズムの値に近くなることが理想であると考えた。提案アルゴリズムは、閲覧履歴に基づくアルゴリズムで並べられた記事中、ある割合でユーザーと嗜好の異なる記事の中で注目されている記事で置き換えることによって作成した。置き換えを行う割合を変えながら、提案アルゴリズムによるシミュレーションを行なった結果、閲覧履歴に基づくアルゴリズムにより並べた記事の9%をユーザーの嗜好と異なる記事で置き換えた場合において提案アルゴリズムの理想を達成することができた。しかし、この結果は、本研究のモデルによるシミュレーションのパラメーターに依存するもので、実際のSNSにおいて、置き換える割合が9%でよいということは明らかにはしていない。実際のSNSでは、ユーザー数やステップ数などは指定ができないため、最適値を見つけるための検証は、実際に運用されているサービス内で行う必要がある。したがって、閲覧履歴に基づくアルゴリズムで並べられた記事の一部を、ユーザーの嗜好と異なる記事で置き換えることが証明されたことが、本研究の意義であるといえる。

本研究において、ユーザーにとって有益な情報を見つけやすく、かつ、意図しないフィルターバブルからユーザーを守るためには、閲覧履歴に基づくアルゴリズムで並べられた記事の一部を、ユーザーの嗜好と異なる記事で置き換えればよいということが明らかになった。今後、実際のSNSでの検証により、その効果の証明がされることを期待する。また、今回作成したモデルは、共クラスタリングが行われていることを前提としているため、ユーザーに記事を推薦する手法による効果の違いを検証することはできない。今後、SNSの運営側が利用できる、よりよいアルゴリズムを開発するための総合的なシミュレーション環境が構築されることを期待する。

## 参考文献

- 1) Eli Pariser: The Filter Bubble: What The Internet Is Hiding From You, Penguin, UK(2011)
- 2) 鳥海不二夫, 山本 仁志: ソーシャルメディアにおける協調の進化, 情報処理学会論文誌, **53-11**, 2507/2515(2012)
- 3) 三ヶ尻陽一: 同調圧の性質を考慮に入れた二次元 Ising Model による炎上現象の再現とその抑制策の提案, 電子情報通信学会論文誌, **J100-D-1**, 28/35(2017)

- 4) 桑田周平, 山田武士, 上田修功: ディリクレ過程混合モデルに基づく離散データの共クラスタリング, 情報処理学会論文誌, **1-1**, 60/73(2008)
- 5) 神鷲敏弘, 推薦システムのアルゴリズム(1): 人工知能学会論文誌, **22-6**, 826/837(2007).
- 6) 土方嘉徳, 嗜好抽出と情報推薦技術: 情報処理学会論文誌, **48-9**, 957/965(2007)