

# HappyRec: 幸せ推薦システムの実装及び評価手法の検討

鈴木 健太<sup>†</sup> 田中涼太郎<sup>††</sup> Panote Siriaraya<sup>†††</sup> 栗 達<sup>††††</sup> 中島 伸介<sup>††††</sup>

<sup>†</sup> 京都産業大学大学院 先端情報学研究科 〒603-8555 京都府京都市北区上賀茂本山

<sup>††</sup> 京都産業大学 コンピュータ理工学部 〒603-8555 京都府京都市北区上賀茂本山

<sup>†††</sup> 京都工芸繊維大学 情報工学・人間科学系 〒606-8585 京都市左京区松ヶ崎橋上町

<sup>††††</sup> 京都産業大学 情報理工学部 〒603-8555 京都府京都市北区上賀茂本山

E-mail: <sup>†</sup>i2086060@cc.kyoto-su.ac.jp, <sup>††</sup>g1744717@cc.kyoto-su.ac.jp, <sup>†††</sup>spanote@kit.ac.jp,

<sup>††††</sup>{lida,nakajima}@cc.kyoto-su.ac.jp

あらまし 心理学者 Seligman らが提唱するポジティブ心理学において、毎晩その日の幸せな出来事を 3 つ記録することにより幸福度の向上と抑うつ度の低下が起こるという研究が報告されている。しかし、日常生活で自覚できる幸せな出来事は決して多くはない。そこで、幸せな出来事を推薦することで、経験する幸せの回数を増やし、幸福度向上、抑うつ度の低下を狙えるのではないかと考えた。本稿では幸せ推薦システムの構築方法とシステム内で行う協調フィルタリングの予備実験、および提案システムに基づく評価実験結果について報告する。

キーワード 幸せ推薦システム, Web マイニング, 幸福度, ウェルビーイング, 地理情報, HappyDB

## 1. はじめに

心理学者 Seligman らが提唱するポジティブ心理学 [1] において、three good things という、毎晩その日の幸せな出来事を 3 つ記録するエクササイズが存在する。このエクササイズを行うことによって、幸福度の向上と抑うつ度の低下が起こると報告 [2] されている。この研究に関して、関沢らの研究 [3] で日本人被験者による実験を用いて、同様の結果が報告されている。しかし、このエクササイズでは自力でその日の幸せな出来事を思い出す必要があるが、自尊心の低い人やうつ病の人は普段の生活の中で幸せな出来事を自覚することは容易ではない。図 1 は厚生労働省が公開している患者調査 [4] の結果で、うつ病患者数の推移を示している。調査が開始された平成 8 年から平成 28 年で 3 倍ほどになっていることが分かる。これらから、幸せな出来事を推薦することで、経験する幸せの回数を増やし、幸福度の向上、抑うつ度の低下を狙えるのではないかと考え、幸せ推薦システムの開発を開始した。本稿では幸せ推薦システムの構築方法とシステム内で行う協調フィルタリングの予備実験、提案システムの評価実験の計画について説明する。

本稿の構成は以下の通りである。2 章では関連研究を紹介する。3 章では本稿で提案するシステムの概要を解説する。4 章では協調フィルタリングを使用した幸せ情報の追加についての予備実験と結果について報告する。5 章では幸せスポット推薦の評価実験について説明し、最後に 6 章でまとめを記述する。

## 2. 関連研究

本稿で提案するシステムは、心理学者の Seligman らが提唱するポジティブ心理学 [1] から着想している。この Seligman らの研究では、three good things という毎晩寝る前にその日あった幸せな出来事を 3 つその理由とともに書くエクササイズがある。このエクササイズを行うことで幸福度の向上と抑うつ

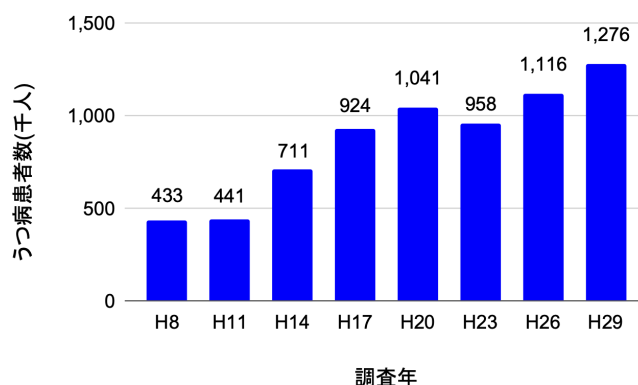


図 1 うつ病 (躁鬱病含む) 患者数の推移 (日本国内)

度の低下が起き、この効果の持続性を報告 [2] している。また、Seligman らの実験に関して、関沢ら [3] は日本人の被験者で同様の実験を行い、幸福度が向上することを報告している。これらの研究では日頃起きる幸せな出来事を認知して記録する必要があるが、現実には難しい。そのため、本稿で提案する幸せ推薦システムによって身近で経験できる幸せな出来事を推薦することは有用だと考える。

幸せに関する研究としては、幸せの瞬間を収集しデータベース化する試みとして Asai らの HappyDB [5] がある。これは欧米人の幸せな瞬間をクラウドソーシングにより約 10 万件のテキストデータを収集したものであり、人々がどのような瞬間に幸せを感じるのかを分析する上で貴重なデータである。本稿では、このデータセットを使用して提案システムの構築を行う。詳細は 3 章で説明する。

また、ユーザの感情を重視した情報推薦技術の研究として、Quercia らの研究 [6] が存在する。この研究はソーシャルデータ (Flicker のメタ付き情報) を使用し、最短経路ではなく、ユー

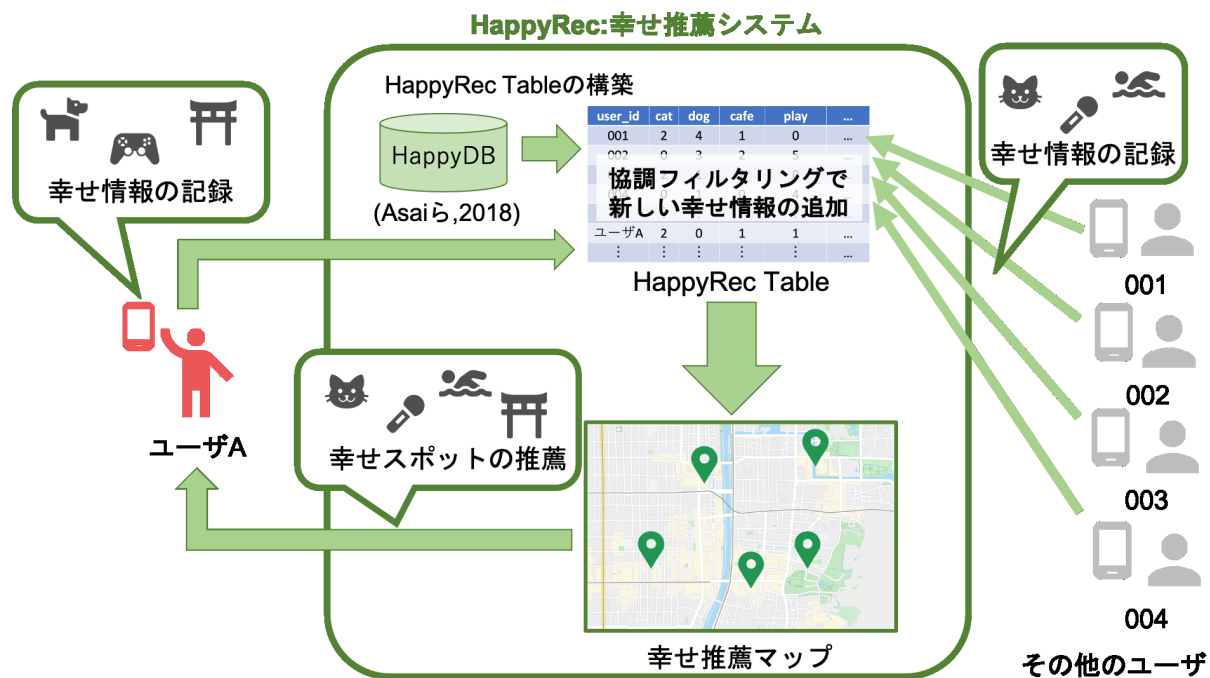


図 2 幸せ推薦システムの概要

ザが静かで、美しく、幸せに感じるであろうルートを紹介するというものである。しかし、この研究では、美しさ、静かさなどの全てのユーザーに当てはまる感情をもとに推薦を行っている。本稿で提案する手法では、ユーザーそれぞれに特化した推薦が可能になると考えられる。

これら以外にも、Yin らの研究 [7] では、位置情報に注目した推薦システムを提案している。この研究では、ユーザーの活動履歴を用いて、ユーザーの興味のあることを抽出し、これをもとにイベントや場所の推薦を行っている。また、ユーザーの興味だけではなく、ユーザーにとって新しい都市を推薦する際は、その場所特有のイベントや場所を考慮することによって推薦の精度向上を図っている。この研究ではユーザーの行動履歴から興味のあるものを推薦しているが、本稿で提案する手法ではユーザーの幸せな体験の情報を用いることは斬新な視点だと考えている。

### 3. 幸せ推薦システムの開発

本章では、3.1 節で幸せ推薦システムの概要、3.2 節で HappyRec Table の作成について、3.3 節で幸せな体験ができる場所の推薦方法について説明する。

#### 3.1 幸せ推薦システムの概要

本節では幸せ推薦システムの概要を図 2 を用いて説明する。

図左のユーザー A が幸せなスポットを推薦されるユーザーである。提案システムでは、利用ユーザーが日頃感じる幸せな出来事を日記感覚でシステムに記録する。システムは投稿された幸せな情報を HappyRec Table として記録する。この記録された情報を使用して協調フィルタリングを用いることで、まだ経験したことのない未知の幸せな情報を追加する。この未知の情報も追加された HappyRec Table と類似度の高いスポットをユーザーが幸せな体験のできるスポットとして提示する。

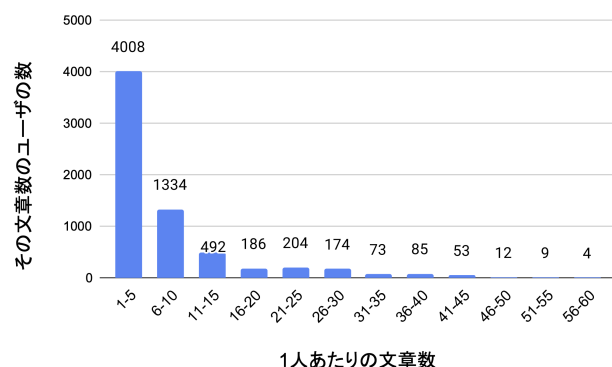


図 3 HappyDB の 1 人あたりの幸せな出来事の記録回数

#### 3.2 HappyRec Table の作成

提案システムで使用する HappyRec Table の構築方法を説明する。3.1 節で書いた通り、提案システムでは HappyRec Table を用いて協調フィルタリングで新しい幸せ情報の推薦を試みる。そのため、実際の利用ユーザーの情報のみで構築した場合、新たな幸せ情報をユーザーに推薦することが困難になるコールドスタート問題が生じることが考えられる。これを解決するため、予め HappyRec Table に一定量の幸せ情報を登録しておく必要がある。今回は Asai らの公開している HappyDB [5] を使用する。HappyRec Table の構築の前に HappyDB について説明する。

HappyDB はクラウドソーシングによって集められた欧米人の幸せな瞬間を説明したテキストデータから構成されている。データは約 1 万人から集められ、全部で約 10 万件の文章が記録されている。文章ごとに、その文章の幸せな出来事が、記録作業を行った時間から 24 時間以内のものの、3ヶ月以内のもの

表 1 HappyRec Table

| userid | cat | dog | cafe | play | ... |
|--------|-----|-----|------|------|-----|
| 001    | 2   | 4   | 1    | 0    | ... |
| 002    | 2   | 0   | 1    | 1    | ... |
| 003    | 4   | 3   | 1    | 4    | ... |
| :      | :   | :   | :    | :    | ... |
| Suzuki | 0   | 3   | 2    | 5    | ... |
| :      | :   | :   | :    | :    | ... |

|              | user_id | cat | dog | cafe | play | ... |
|--------------|---------|-----|-----|------|------|-----|
| 比較対象の<br>ユーザ | 001     | 2   | 4   | 3    | 0    | ... |
|              | 002     | 2   | 0   | 1    | 1    | ... |
|              | :       | :   | :   | :    | :    | ... |
| 推薦対象の<br>ユーザ | Suzuki  | 3   | 0   | 3    | 1    | ... |
|              | Tanaka  | 0   | 0   | 0    | 1    | ... |
|              | :       | :   | :   | :    | :    | ... |

図 4 未知の幸せ単語の追加

かといった情報も付いている。

HappyRec Table の構築には HappyDB に入っている 24 時間以内の幸せについて説明した文章のみ使用する。直近 3ヶ月以内の幸せについてのデータではなく、24 時間以内のデータのみを使用した理由は、直近 3ヶ月以内の幸せとして記録されているものは、ユーザにとって長期間忘れない大きな出来事が多く記録されており、直近 24 時間以内の幸せに関しては、その日の出来事であり、毎日起きる小さな幸せが記録されていると考えたためだ。HappyDB には 24 時間以内の幸せについて説明した文章のデータは 6,634 人、49,831 文章存在する。このデータの内、ユーザ 1 人あたりの幸せな出来事の記録回数を集計したところ、図 3 のようになった。記録回数が少ないユーザが多いことが分かる。提案システムでは毎晩記録してもらうことを想定しており、1 人あたりのデータ量が多くなることが想定されるため、HappyDB から使用するデータでは 1 人で 30 文章以上記録している 309 人のデータを使用することにした。文章数としては 11,311 文章になる。

次に HappyRec Table について説明する。HappyRec Table の例を表 1 に示す。これは、ユーザの識別 ID とユーザが投稿した幸せの単語ごとの出現回数を記録したデータであり、表 1 の時は、ユーザ 001 の幸せ記録に cat が 2 回、dog が 4 回、cafe が 1 回出現し、play は出現していないことを示している。

構築方法としては、上で説明した HappyDB のテキストデータを文章ごとに形態素解析し、前置詞や冠詞などの stopwords を抜いた単語の出現回数とユーザの識別 ID を使用して HappyRec Table を作成する。

### 3.3 幸せな体験のできる場所推薦

3.1 節、3.2 節で作成した HappyRec Table を用いて、ユーザが幸せな体験のできる場所の推薦方法について説明する。

推薦には一部、協調フィルタリングを用いる。具体的な方法は図 4 を用いて説明する。図中のユーザ ID001~002 のデータは HappyRec Table 構築時に使用した HappyDB のユーザで

表 2 正規化無しデータの上位 10 件に対する予測結果

| ユーザ ID | 単語        | 正解値 | SVD 予測 | SlopeOne 予測 |
|--------|-----------|-----|--------|-------------|
| 230    | life      | 100 | 4.09   | 1.87        |
| 120    | event     | 52  | 1.73   | 0.46        |
| 6      | Yesterday | 36  | 1.21   | 0.71        |
| 74     | life      | 36  | 4.42   | 2.07        |
| 40     | home      | 33  | 2.59   | 1.29        |
| 39     | Yesterday | 32  | 2.03   | 0.75        |
| 40     | Yesterday | 32  | 2.56   | 0.89        |
| 53     | research  | 32  | 3.2    | 0.19        |
| 132    | life      | 32  | 1.63   | 2.08        |
| 74     | moment    | 31  | 6.76   | 1.76        |

表 3 正規化無しの予測精度

|      | SVD    | SlopeOne |
|------|--------|----------|
| RMSE | 43.546 | 45.160   |
| MAE  | 38.578 | 40.393   |

あり、Suzuki は実際の利用ユーザで、今回幸せな体験のできる場所を推薦されるユーザとする。Suzuki が既に記録している単語のみを使用した推薦では、新しい幸せの推薦ができなく、過去に記録したことの無い幸せも推薦可能にするために協調フィルタリングを使用する。推薦されるユーザである Suzuki と類似度の高いユーザの幸せ情報の中から、Suzuki に存在しない項目を新たに Suzuki の幸せに関連の深い単語として、既に記録された単語に追加して推薦を行う。この作業には Python のライブラリである Surprise [9] を使用する。

以上の過程では、HappyRec Table を用いてユーザの幸せと関連の深い単語の推薦までを行った。次に、この幸せに関連のある単語とユーザの周辺のスポット情報を使用して幸せを感じるであろうスポットの推薦方法を説明する。

ユーザの現在地周辺のスポットを OpenStreetMap [10] や Google マップといったサービスから複数箇所選ぶ。このスポットごとに付与されているスポットのカテゴリや口コミ情報などの単語を Word2vec を用いてベクトル化する。同様にユーザの幸せに関連の深い単語を Word2vec を用いてベクトル化する。この際スポットごと、ユーザごとに単語を足し合わせることでそれぞれ 1 つのベクトルになるようにする。ユーザのベクトルと周辺のスポットごとのベクトルとの間でコサイン類似度を求め、類似度の高いスポット上位数件をユーザが幸せを感じるスポットとしてマップ上に表示して推薦を行う。

## 4. 予備実験

3.3 節で説明した過去に記録したことの無い幸せを協調フィルタリングで追加する処理について、3.2 節で作成した HappyRec Table を用いて予備実験を行った。具体的には、HappyRec Table の値が大きい順に 10 箇所の値をそれぞれ、0 に変更して未知の値にし、これらの値の予測を SVD と SlopeOne を用いて行った。この予測には Python のライブラリである Surprise [9] を使用した。

今回は、HappyRec Table の値をそのまま使用したものと、



表 4 正規化ありデータの上位 10 件に対する予測結果

| ユーザ ID | 単語        | 正解値  | SVD 予測 | SlopeOne 予測 |
|--------|-----------|------|--------|-------------|
| 230    | life      | 1    | 0.02   | 0           |
| 120    | event     | 0.52 | 0      | 0           |
| 6      | Yesterday | 0.36 | 0.01   | 0           |
| 74     | life      | 0.36 | 0.02   | 0           |
| 40     | home      | 0.33 | 0.01   | 0           |
| 39     | Yesterday | 0.32 | 0.01   | 0           |
| 40     | Yesterday | 0.32 | 0.01   | 0           |
| 53     | research  | 0.32 | 0      | 0           |
| 132    | life      | 0.32 | 0.02   | 0           |
| 74     | moment    | 0.31 | 0.02   | 0           |

表 5 正規化ありの予測精度

|      | SVD   | SlopeOne |
|------|-------|----------|
| RMSE | 0.452 | 0.463    |
| MAE  | 0.404 | 0.416    |

HappyRec Table の最小値と最大値を使用して正規化した 2 種類のデータで予測作業を行った。正規化無しのデータでの予測結果を表 2 と表 3 に示す。正規化したデータでの予測結果を表 4 と表 5 に示す。

これらの結果から、予測精度が良くないことが分かる。しかし、本稿で提案するシステムでは 3.3 節で説明した通り、幸せな体験のできるスポットを推薦することを目指している。そのため、単語の予測性能より最終的に推薦するスポットの精度が重要なため、単語の予測性能の向上は今後の課題とし、本稿では 3 章で説明した手法での推薦を実施し性能の評価を行うことにした。

## 5. 評価実験

### 5.1 評価実験の概要

3 章で説明した幸せ推薦システムが実際にユーザが幸せを感じるスポットを推薦することが可能かどうかを検証する被験者実験を行う。

大学生 9 人の被験者に 2 週間、毎晩寝る前にその日あった幸せな出来事を 3 つ記録してもらった。記録の方法は、先行研究 [8] で開発しているスマートフォンアプリと Google スプレッドシートを使用して行った。この記録で収集された幸せな出来事を記録したテキストデータを使用して被験者ごとに幸せを感じるであろうスポットの推薦を実施し、nDCG を用いて評価する。

### 5.2 評価実験の流れ

最終的な評価の nDCG で使用する正解のランキングを作成するために、独自で作成した京都市内の 20 スポットを幸せに感じる順番に並び替えてもらうアンケートを実施した。このアンケートは各スポットごとにスポットの名称、Google マップのストリートビューへのリンク、スポットのジャンルを記載することで、こういった場所かを被験者に分かりやすく示した。このアンケートを 2 週間の実験の初日に回答してもらう。このアンケートで収集されたランキング情報を nDCG の正解ランキ



図 5 日本語データでの WordCloud



図 6 英語データでの WordCloud

ングとして使用する。

正解ランキングとは別に、被験者が 2 週間記録した幸せな出来事の情報を 3 章で説明した提案システムで分析することで、被験者ごとに幸せを感じるであろうスポットの予想ランキングを作成する。これらのランキング情報を使用して nDCG を求めて提案システムの評価を行う。

### 5.3 収集された幸せ情報について

被験者 9 人から集まった幸せな出来事を説明したテキストデータについて説明する。被験者には 2 週間毎晩寝る前にその日あった幸せな出来事を、出来事の説明と、それがなぜ幸せに感じたのかの説明のペアを 3 つ記録してもらった。日によって記録を忘れることもあったため、2 週間で合計 347 セットの記録が集まった。集まった文章データから WordCloud を作成したものを図 5 に示す。また、提案システムでは英語に翻訳した状態でも試すため収集された幸せ情報を Google 翻訳にかけ、翻訳語のデータでの WordCloud を図 6 に示す。

### 5.4 幸せを感じるスポットの推薦結果

収集したデータを用いて 3.3 節で説明した手法で実際に被験者へのスポットの推薦を実施した。推薦は、日本語データと Google 翻訳で英訳したデータの 2 種類で行い、日本語データでは形態素解析器に MeCab を使用し、単語の分散表現の取得に

表 6 日本語データでの各手法ごとの nDCG の平均値

| 手法                      | Top5<br>単語使用 | Top10<br>単語使用 | Top20<br>単語使用 | Top30<br>単語使用 |
|-------------------------|--------------|---------------|---------------|---------------|
| SVD<br>(変換無し)           | 0.793        | 0.789         | 0.775         | 0.776         |
| SlopeOne<br>(変換無し)      | 0.802        | 0.781         | 0.776         | 0.775         |
| SVD<br>(最大値で正規化)        | 0.780        | 0.776         | 0.777         | 0.781         |
| SlopeOne<br>(最大値で正規化)   | 0.798        | 0.801         | 0.776         | 0.774         |
| SVD<br>(z 値で正規化)        | 0.796        | 0.787         | 0.778         | 0.774         |
| SlopeOne<br>(z 値で正規化)   | 0.790        | 0.796         | 0.782         | 0.776         |
| SVD<br>(tfidf に変換)      | 0.803        | 0.811         | 0.798         | 0.796         |
| SlopeOne<br>(tfidf に変換) | 0.786        | 0.783         | 0.774         | 0.774         |

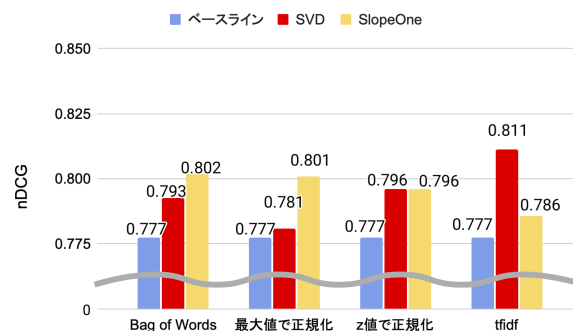


図 7 日本語データの各手法の結果の比較

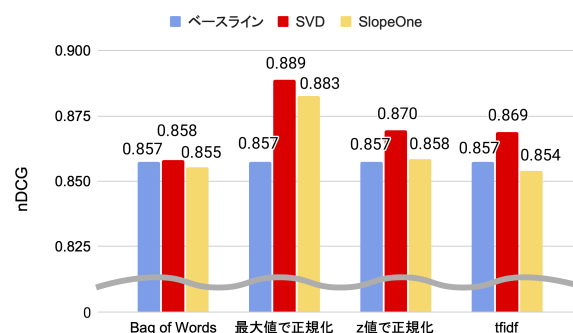


図 8 英語データの各手法の結果の比較

表 7 英語データでの各手法ごとの nDCG の平均値

| 手法                      | Top5<br>単語使用 | Top10<br>単語使用 | Top20<br>単語使用 | Top30<br>単語使用 |
|-------------------------|--------------|---------------|---------------|---------------|
| SVD<br>(変換無し)           | 0.837        | 0.851         | 0.850         | 0.858         |
| SlopeOne<br>(変換無し)      | 0.839        | 0.855         | 0.848         | 0.853         |
| SVD<br>(最大値で正規化)        | 0.889        | 0.857         | 0.860         | 0.861         |
| SlopeOne<br>(最大値で正規化)   | 0.883        | 0.853         | 0.864         | 0.856         |
| SVD<br>(z 値で正規化)        | 0.870        | 0.865         | 0.852         | 0.852         |
| SlopeOne<br>(z 値で正規化)   | 0.856        | 0.856         | 0.852         | 0.858         |
| SVD<br>(tfidf に変換)      | 0.856        | 0.866         | 0.865         | 0.869         |
| SlopeOne<br>(tfidf に変換) | 0.823        | 0.854         | 0.851         | 0.848         |

表 8 協調フィルタリングを使用しない結果

| 手法                | nDCG  |
|-------------------|-------|
| 日本語 (協調フィルタリング無し) | 0.777 |
| 英語 (協調フィルタリング無し)  | 0.857 |

は Word2vec を使い、Word2vec のモデルは白ヤギコーポレーションが公開しているものを使用した。英語データでは形態素解析器には nltk を使用し、単語の分散表現の取得には Word2vec を使い、Word2vec のモデルは Google ニュースで学習されたものを使用した。日本語、英語のデータで単語の出現回数を記録した HappyRec Table を作成し、作成した HappyRec Table から最大値を用いた正規化、標準偏差に基づく z 値を用いた正規化、また、(1) 式 (2) 式を用いた、通常の tfidf 算出の文書頻度をユーザ頻度とした値のそれぞれに変換した HappyRec Table を作成した。

$$tf(t_i, u_j) = \frac{\text{ユーザ } u_j \text{ が単語 } t_i \text{ を記述した回数}}{\text{ユーザ } u_j \text{ のすべての単語の出現回数の和}} \quad (1)$$

$$idf(t_i) = \log\left(\frac{\text{総ユーザ数}}{\text{単語 } t_i \text{ を記述したユーザ数} + 1}\right) \quad (2)$$

作成したそれぞれの HappyRec Table で Surprise を使用し SVD と SlopeOne による協調フィルタリングを実施、協調フィルタリングによるスコアが高い順に 5 単語、10 単語、20 単語、30 単語の合計ベクトルをそれぞれ求め、これらを被験者のベクトルとしてスポットとの類似度を求めて類似度の高い順に被験者が幸せを感じるであろうスポットのランキングを作成した。作成したランキングと 5.2 節で作成した被験者による正解ランキングの間で nDCG を求めた。日本語データで実施した nDCG の結果を表 6 に示し、それぞれの手法で一番良い結果をまとめたものを図 7 に示す。英語データで実施した nDCG の結果を表 7 に示し、それぞれの手法で一番良い結果をまとめたものを図 8 に示す。また、ベースラインとして協調フィルタリ

ングを使用せず、被験者ごとの出現単語全ての合計ベクトルを被験者のベクトルとする手法を行い、この結果を表8に示す。

図7と図8から、ベースラインである協調フィルタリングを行わない推薦結果と比べ、HappyRec Tableを正規化などで変換し、協調フィルタリングを行った推薦方法の方が優れた幸せスポットランキングを作成することが可能だということが分かった。この理由としては、ベースラインでは幸せ記録の単語を全て使用してスポットとの類似度を求めており、被験者の幸せな出来事と関係の浅い単語も使用している。しかし、協調フィルタリングを利用する提案手法では、協調フィルタリングのスコアの上位単語のみを使用してスポットとの類似度を求める。これにより、被験者の幸せと関係の深い単語のみを利用することが可能となり、被験者が幸せに感じるスポットを的確に見つけることができたと考えられる。

また、図7と図8から日本語データの時は単語の出現回数を記録したHappyRec Tableをtfidfで変換し、SVDで協調フィルタリングを行った結果がより良くなり、英語データの時は最大値で正規化し、SVDで協調フィルタリングを行った結果がより良くなったことが分かる。

## 6. おわりに

本稿では、ユーザが幸せな体験ができる場所を推薦する方法について説明し、提案システム内で行う協調フィルタリングをHappyDBのデータで実施し精度を確認した。また、2週間の実験を行い、提案システムで作成するスポットのランキングの評価をnDCGを用いて実施した。

今後は、提案システムで作成するランキング精度の向上と、より気軽に使用可能にするために、スマートフォンアプリの開発、より大人数での実験での評価を考えている。

## 謝 辞

本研究の一部は、科研費基盤研究(B)(課題番号:17H01822, 19H04118, 20H04293)による。ここに記して謝意を表す。

## 文 献

- [1] Seligman, M.E.P. (2002). Authentic Happiness: Using the New Positive Psychology to Realize Your Potential for Lasting Fulfillment. The Free Press.
- [2] Martin E. P. Seligman and Tracy A. Steen, Nansook Park, Christopher Peterson. Positive psychology progress: Empirical validation of interventions, American Psychologist(60,410–421), 2005.
- [3] 関沢 洋一, 吉武 尚美. 良いことを毎日3つ書くと幸せになれるか?, RIETI Discussion Paper Series, 2013年11月.
- [4] 厚生労働省, 患者調査 (<https://www.mhlw.go.jp/toukei/list/10-20.html>)
- [5] Akari Asai, Sara Evensen, Behzad Golshan, Alon Halevy, Vivian Li, Andrei Lopatenko, Daniela Stepanov, Yoshihiko Suhara, Wang-Chiew Tan, Yinzhan Xu, HappyDB: A Corpus of 100,000 Crowdsourced Happy Moments, 2018.
- [6] Quercia, D., Schifanella, R., and Aiello, L. M. (2014, September). The shortest path to happiness: Recommending beautiful, quiet, and happy routes in the city. In Proceedings of the 25th ACM conference on Hypertext and social media (pp. 116-125). ACM.
- [7] Yin, H., Sun, Y., Cui, B., Hu, Z., and Chen, L. (2013, Au-

gust). LCARS: a location-content-aware recommender system. In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 221-229). ACM.

- [8] 田中涼太郎, Panote Siriaraya, 鈴木 健太, 中島 伸介. Happy-Button:日々の幸せ体験認識システムの開発. DEIM2020 P1-8.
- [9] Surprise, (<http://surpriselib.com>)
- [10] OpenStreetMap, (<https://openstreetmap.jp/>)