

映画検索のための特徴表現フレーズ選択に基づく適合性フィードバック

田巻 優作[†] 北山 大輔[†]

[†] 工学院大学情報学部 〒163-8677 東京都新宿区西新宿 1-24-2
E-mail: tj115070@ns.kogakuin.ac.jp, ††kitayama@cc.kogakuin.ac.jp

あらまし 近年映画は増え続け、これまでに公開された映画の数は6万本を越える。ユーザが映画を検索する場合、通常は監督や俳優、カテゴリなどを用いることが多い。しかし、このような検索ではより細分化された特徴をユーザが入力することは難しく、ユーザの検索意図を正確に反映することができない。そこで我々は、映画の細かい要素を説明する特徴表現フレーズに対し、ユーザが好みを入力することで検索クエリを生成し、適合する映画を検索することを繰り返す適合性フィードバック手法を提案する。本稿では、レビュー中から映画の特徴を表す短い文を特徴表現フレーズとして抽出し、ユーザの要求を表現するために利用する。また、プロトタイプシステムを構築し、抽出した特徴表現フレーズの評価および、適合性フィードバックによる検索精度を評価する。

キーワード 映画検索, 分散表現, 適合性フィードバック

1 はじめに

日本映画製作者連盟¹によると国内で一年間に公開される映画の本数は2017年では邦画・洋画合わせて1187本となり、近年この数値は増加傾向にある。また、Yahoo!映画²によると、これまでに公開された映画の総数は2018年現在で6万本を越えている。そのような状況下で、詳しい知識を持たないユーザが、6万本の映画の中から自分の要求に合った映画を見つけることは困難である。また昨今は定額の動画配信サイトが普及し、過去に公開された映画を誰でも手軽に視聴できる環境が整いつつある。このような状況下で映画の推薦や検索の重要性はますます高くなってきている。これに対してさまざまな検索や推薦の手法が実用化されている。その方法はユーザの視聴履歴を元にしたもの[1]や、コンテンツの情報を元にしたもの[2]など多岐に渡るが、それぞれに特徴があり一長一短である。その中の一部については2章の関連研究で述べる。

本研究では手軽に行える映画検索を目標とする。ユーザが映画の一覧に目を通すことやレビューの評価、またレビューの入力などは、システムが映画を探す上で有益な情報源となりえるが、ユーザにとっては負担になる。また視聴履歴などのユーザに関するデータは、特定のサービスを長く使っていないならば十分なデータが集まらないという問題がある。そのため事前情報を必要とせず、ユーザが簡単な作業を行うだけで、要求にあった映画の情報を得ることができるシステムがあれば有益であろうと考えた。

そこで我々は映画に詳しくないユーザが自分の要求を明確にしていく段階を踏んで、要求を満たす映画を見つけられるようなシステムの構築を目指した。ここで想定されるユーザは映画に対して専門的な知識を有さず、一本の映画を探すことにあまり手間を掛けたくない人物である。よって提案手法は誰でも容

易に扱え、手軽に映画を検索できることが要件となる。

提案手法は大きく分けて前処理と映画検索の二つの段階に分けられる。前処理の段階では、映画の部分的な特徴を表す短い文をレビュー中から抽出する。本稿ではこの文のことを特徴表現フレーズと呼称する。特徴表現フレーズの抽出は、形態素解析を行い、単語のTF-IDF値を求めて文の特徴量を算出することで行う。またこれによって得られたスコアに従って、互いに類似せず、より特徴を表現している文を抽出する。最終的に選ばれた文が特徴表現フレーズである。

映画検索の段階では適合性フィードバックによって、ユーザの要求にあった映画を検索する。まずユーザに幾つの特徴表現フレーズを提示し、ユーザはそれが自分の要求に合ったものかを評価する。その結果によって検索クエリを作成し、個別の映画が持つ特徴表現フレーズと比較して検索を行う。その後、新たな特徴表現フレーズを提示することでクエリの修正を行う。本稿では特徴表現フレーズ抽出と検索システムについて評価を行う。図1は、検索システムのユーザインタフェースである。

2章では関連研究、3章では特徴表現フレーズ抽出、4章では適合性フィードバックによる検索、5章では特徴表現フレーズの評価実験、そして6章では検索システムの評価実験について述べる。

2 関連研究

2.1 映画の推薦・検索

インターネット上で膨大な数の映画を視聴できる昨今、映画の推薦や検索に関する研究は数多く存在する。

映画レビューから推薦文を抽出する手法として石田ら[3]の研究がある。ユーザの嗜好を表したタグを用いてレビュー内の文を検索し、肯定的な評価を下している文をユーザに提示する手法である。レビュー内の一文を抜粋する点において本稿の研究と類似している。

ユーザによって入力されたレビューを用いて推薦を行う手法

1 : <http://www.eiren.org/toukei/data.html>

2 : <https://movies.yahoo.co.jp/movie/>

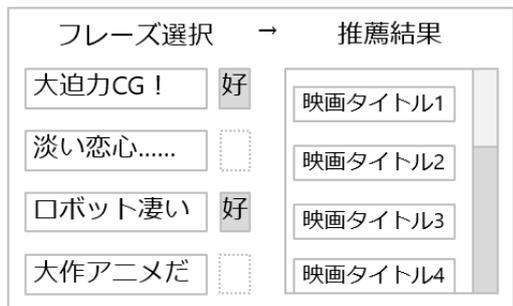


図1 特徴表現フレーズ選択に基づく適合性フィードバック

には林ら [4] の研究がある。この手法ではレビューの中からユーザが興味を持っている評価項目を抽出し、推薦候補の映画のレビューと比較することで推薦を行う。

映画に付けられたレビューから映画間の関連性を導き出し、ユーザ好みの映画を推薦する手法としては櫛見ら [2] の研究がある。入力された映画との類似度を、全体的な観点と部分的な観点に分けて求めることで映画を推薦した。

ユーザによって書かれたレビューを用いて類似ユーザを求め、映画を推薦する手法としては岡田ら [1] の研究がある。この研究で提案されているのは、ユーザ間型での協調フィルタリングを、ベクトル化した映画レビューによって行う手法である。

キャッチコピーデータベースと、レビュー中の単語を利用して映画のキャッチコピーを作成する手法として、幅野ら [5] の研究がある。既存のキャッチコピー内の単語を意味素性を用いて置き換え、評価することで新たなキャッチコピーの作成を行う手法である。

本稿の研究は映画という、ストーリーを含んだ娯楽コンテンツの特徴を抜き出す。類似した研究として漫画の細かい内容に基づいた表現を行う山下ら [6] の研究がある。この研究では TF-IDF と LDA を用いてレビュー文から内容情報を示す複数の単語を取得する手法を提案した。

2.2 ショートテキスト処理

昨今の情報推薦・検索においてテキスト処理はごく一般的な手法であり、本研究でも TF-IDF や word2vec が使われている。中でもショートテキストとしてよく研究対象になる Twitter については、以下のような研究が存在する。

久米ら [7] は Twitter のユーザ推薦において、被推薦者の嗜好のカテゴリを TF-IDF 値を使って割り出した。通常の TF-IDF では類似した嗜好を表しても別々に評価されてしまうが、それらをまとめて評価することでより正確な嗜好を抽出する手法を提案した。

同じく Twitter でも文と単語の類似度によって、単語の重みを決定して特徴量を算出する田中ら [8] の研究がある。doc2vec を用いて文ベクトルと単語ベクトルを比較することで重みづけを行い、類似トピックにおいてより正確なランキングを可能とした。

2.3 適合性フィードバック

検索の精度を高める方法として、適合性フィードバックがあげられる。検索結果に対するユーザのフィードバックから再検索を行っていく方法である。適合性フィードバックを使って、検索や推薦の精度を向上させる研究について述べる。

ユーザがレビューを選ぶことで、ユーザと類似した評価基準の重みを増して協調フィルタリングを行う手法として南ら [9] の研究がある。評価観点に関するトピックモデルを作成し、トピック内の単語との一致度を利用してレビュー者のトピック所属率を算出し、ユーザがレビューにフィードバックすることで評価基準を算出して書籍を推薦した。

適合性フィードバックの中でも、上記のように明示的にユーザの嗜好を取得する場合もあれば、行動履歴などから暗黙的に取得する場合もある。杉山ら [10] はユーザの web ページ閲覧履歴からユーザプロフィールを作成し、検索結果の個人化を行った。

3 特徴表現フレーズ抽出

本節では特徴表現フレーズの抽出について述べる。特徴表現フレーズはユーザへの提示、それを読んだユーザによるフィードバック、そしてユーザの要求する特徴と映画自体の特徴として検索に用いられる。特徴表現フレーズに期待される働きは映画を構成する部分的な特徴を表現することである。また、誰でも簡単にというシステムの目標のために、映画に対する専門的な知識を必要としないことと、読みやすい分量であることが求められる。これらを満たした文を抽出する方法について以下に記す。

3.1 特徴表現フレーズのスコアリング

特徴表現フレーズは、映画レビュー中の各文にスコアを付けることによって評価し、抽出する。具体的なスコアリングの手法について詳細を記す。

評価の対象とするのはレビュー中で、句点やエクスクラメーションマークで区切られた一文であり、これらが特徴表現フレーズの候補となる。これらの文を形態素解析器 MeCab [11] によって単語ごとに分解し、各単語の TF-IDF 値を求める。TF-IDF 値は以下の式 (1) によって定義される。式中で、 $n_{w,m}$ は映画 m のレビュー中での単語 w の出現回数、 $\sum_k n_{k,m}$ は映画 m のレビュー中の総単語出現回数、 $|D|$ は総映画数、 $|\{d|d \ni w\}|$ はレビュー中に単語 w が出現する映画の総数である。

$$tfidf_{w,m} = \frac{n_{w,m}}{\sum_k n_{k,m}} \times \log \frac{|D|}{|\{d|d \ni w\}|} \quad (1)$$

文のスコアは文中の単語の TF-IDF 値の平均とする。計算式は以下の式 (2) の通りで、 $tfidf_{w,m}$ は式 (1) に従い、 $|W|$ は文 s 中の単語数である。

$$S_{s,m} = \frac{\sum_w tfidf_{w,m}}{|W|} \quad (2)$$

基本的にはこの計算式に従うが、このままでは特徴表現フレーズの使用用途に適さない文が抽出されることが多い。そこ

で、より特徴を簡潔に表している文を抽出するために、以下の条件付けを行う。

(1) 文のスコアリングに利用する単語は、閾値以下の本数の映画のレビューに出現するものに限定する。TF 値が高すぎると IDF 値が低い場合であっても TF-IDF 値が高くなってしまい、「おもしろい」や「映画」などといった、どの映画のレビューにも出てくるような単語がその映画の特徴を表す単語として使われてしまうことがある。そこで、多くの映画のレビューに出現する単語を除外することで、その映画の特徴を表す単語だけを残すことができると考えた。

(2) 文のスコアリングに利用する品詞は代名詞、数詞以外の名詞、動詞、形容詞に限定する。これによって助詞や接続詞をスコアリングの対象から除外し、特徴に関係ない部分でスコアが下がることを防ぐ。

(3) 文のスコアリングに利用する固有名詞は、閾値以上の本数の映画のレビューに出現するものに限定する。これによって、登場人物名や一般的に意味の通じにくい単語によってスコアが上がることを防ぐ。

(4) スコアリングの対象とする文は、上記 (1)(2)(3) の条件を満たす単語を閾値以上の個数含むものとする。文として意味をなさないものや、十分に特徴を表していない文を除外するためである。

(5) スコアリングの対象とする文は、総単語数が閾値以下のものに限定する。長すぎる文はそれを読むユーザの負担となり、手軽な検索と言うシステムの要件から外れてしまい、好ましくないためである。

(6) レビュー者の要望を述べる文を除外するために、「欲しかった」やそれに相当する表現を含む文をスコアリングの対象から除外する。

(7) 疑問や、レビュー者に自信のない情報を含んだ文を除外するために、「？」を含む文をスコアリングの対象から除外する。

(8) レビューを文に分割する段階で生まれる中途半端な文を除外するために、先頭や末尾に読点が存在する文をスコアリングの対象から除外する。また、括弧や鍵括弧を含む文をスコアリングの対象から除外する。これは括弧内に「！」などの分割に用いる記号が含まれると、そこで分割されてしまうが、そもそも括弧を含む文は単語数が多くなり、単語数による除外条件にかかりやすいことから、これらの文を除外しても問題ないと判断した。

3.2 品詞重みを用いた特徴表現フレーズのスコアリング

前節では名詞、動詞および形容詞をすべて同列に扱ったが、品詞によって映画の特徴を表す度合いに違いがあると考えた。たとえば、「ファンタジー」や「魔法」のような名詞は、「戦う」や「美しい」などの動詞や形容詞よりも細分化された特徴を表しているといった具合である。そこで、文のスコアを算出する際に、品詞によって異なる重みを用いる手法を提案する。これによって、重みの低い品詞が文のスコアに与える影響を小さくする。

計算式は以下の式 (3) に従い、 S_w は単語 w のスコア、 $Weight_w$ は単語 w の品詞の重みである。

$$S_{s,m} = \frac{\sum_w tfidf_{w,m} \times Weight_w}{\sum_w Weight_w} \quad (3)$$

3.3 文の類似度算出に基づくフレーズの選択

ここまで文のスコアリングについて記したが、この条件では TF-IDF 値が高い特定の単語を含む文のみが上位を占めてしまう。本システムにおいて特徴表現フレーズは、ユーザ要求の定義と映画の特徴的な要素の定義に使用されるため、広範な特徴を表すことが望まれる。そこでスコアが高く、また互いに類似しない文を選別する必要がある。

文の類似性の判断には word2vec [12] を用いる。word2vec とは文章をコーパスとして、単語のベクトル化を行う学習モデルの構築を行う手法である。実装としてはプログラミング言語 Python のライブラリ gensim を用いた。今回は DB 中のすべての映画レビューをコーパスとして、単語ベクトルの学習を行った。提案手法では、文中の各単語のベクトルに TF-IDF 値をかけた値を平均したベクトルを文ベクトルとして、文ベクトル間の cos 類似度を計算することで文の類似度を求める。ここで s 、 w はそれぞれ文ベクトルと単語ベクトルを、また $tfidf_{w,m}$ は単語 w の TF-IDF 値を意味する。ここで TF-IDF 値は式 (1) によって求められた値を用いる。

$$s = \sum_w w \times tfidf_{w,m} \quad (4)$$

cos 類似度は以下の式によって定義される。

$$\cos(\mathbf{v}_1, \mathbf{v}_2) = \frac{\mathbf{v}_1 \cdot \mathbf{v}_2}{\|\mathbf{v}_1\| \|\mathbf{v}_2\|} \quad (5)$$

特徴の表現とは関係のない部分が、ベクトルに影響を与えないようにするため、3.1 節の (1)(2) の条件を用いて類似度算出を行った。同 3.1 節の (3) も単語に対する条件だが、出現映画数の少ない固有名詞も含めて類似度を算出したいため、ここでは用いない。これによって、より正確に文の特徴に関する類似度が計算できると考えた。

次に、特徴表現フレーズを選ぶ具体的な手順について説明する。初めに、すべての候補の中から最もスコアの高い文を選び特徴表現フレーズとする。その後、選ばれた 1 つ目の特徴表現フレーズとその他の各文との類似度を計算し、類似度が閾値以上の文を候補から除外する。そうして残った文のうち最もスコアの高い文を 2 つ目の特徴表現フレーズとする。また同様に 2 つ目の文との類似度が高い文を候補から除外し、残った文の中から最もスコアの高い文を選ぶ。これらの処理を繰り返すことによって、望む数の特徴表現フレーズを抽出する。

4 適合性フィードバックによる検索

レビュー中から抽出した特徴表現フレーズを用いてユーザの選択に基づく適合性フィードバックを行う。適合性フィードバックは検索結果に対するユーザからのフィードバックに応じて検索クエリを修正していく手法である。提案手法ではユーザに提

示された特徴表現フレーズへの評価によって、検索クエリを作成して検索を行う。その後、ユーザが繰り返し提示されるさまざまな特徴表現フレーズに評価を与えることで、ユーザの要求する映画を見つけることを目指していく。

初めに、ユーザはキーワードや映画のタイトルによって特徴表現フレーズを検索する。検索はすべての映画のすべての特徴表現フレーズの中から行われる。その後、ユーザは提示された特徴表現フレーズを読み、それが自分の要求に合致するか否かをを入力する。この入力によって検索クエリを作成して映画を検索し、1回目の映画が提示される。検索はユーザが選んだ特徴表現フレーズと、映画が保有する特徴表現フレーズの類似度を求めることによって行う。

本稿では検索の際に行う計算について二つの手法を提案し、以下の4.1節と4.2節にて述べる。本文中で述べるベクトルの類似度の計算には前述した word2vec による文の類似度算出手法を用いる。

その後の二回目のユーザ入力ではよりユーザの要求を絞り込むために、また異なる特徴表現フレーズを提示する必要がある。この方法については4.3節にて述べる。

これらの特徴表現フレーズ提示・ユーザ評価・映画検索を繰り返すことによってユーザの要求を明確にし、要求を満たす映画をユーザに提示していく。

4.1 部分ベクトル類似度による適合性フィードバック

特徴表現フレーズは映画の特徴の部分的な構成要素を表している。この特性を利用してユーザの要求と類似した部分を持つ映画を提示することで、より視野の広い映画検索を行うことができる考えた。

まずユーザが特徴表現フレーズを選択し、ここで選ばれた特徴表現フレーズのベクトルを個別に保持する。そして検索の際のスコアではすべての映画に対して、ユーザが選んだある特徴表現フレーズとその映画の保持する各特徴表現フレーズとの類似度の中でもっとも高い値をユーザが選んだある特徴表現フレーズと対象の映画の類似度として保持する。これをユーザが選んだ各特徴表現フレーズについて計算し、これらの総和をユーザの要求と対象映画との類似度とする。これによって得られた類似度の高い映画をユーザに提示する。

式(6)(7)(8)はユーザの嗜好を表した特徴表現フレーズの集合である U と対象の映画が持つ特徴表現フレーズの集合 M の類似度を表す。 U の中にはユーザがこれまでに肯定的に評価したすべての特徴表現フレーズの集合である U_p と、これまでに否定的に評価したすべての特徴表現フレーズの集合である U_n が存在する。 \mathbf{u} はユーザが選んだ特徴表現フレーズのベクトルであり、ユーザが評価した数だけ存在する。 \mathbf{m} は映画 M が持つ特徴表現フレーズのベクトルであり、特徴表現フレーズ抽出時に設定した数だけ存在する。

$$S_{U,M} = S_{U_p,M} - S_{U_n,M} \quad (6)$$

$$S_{U_p,M} = \sum_{\mathbf{u} \in U_p} \max\{\cos(\mathbf{u}, \mathbf{m}) | \mathbf{m} \in M\} \quad (7)$$

$$S_{U_n,M} = \sum_{\mathbf{u} \in U_n} \max\{\cos(\mathbf{u}, \mathbf{m}) | \mathbf{m} \in M\} \quad (8)$$

その後繰り返されるフィードバックに対しても、同様にベクトルを保持する。そのようにしてユーザの要求を示すベクトルを増やしていき、より類似性の高い映画を検索することができる。この計算方法によって期待できる効果は、前述の通り部分的な特徴の一致についてベクトルを単純に加算する場合よりも高く評価できることである。この手法では、単純にベクトルを加算したのでは得られない、正確なユーザの意図を読み取ることができると考えた。

4.2 合算ベクトル類似度による適合性フィードバック

前節では個々のベクトルの類似度の和で最終的な類似度を算出する手法について述べたが、本節では合算したベクトル同士の類似度によって検索を行う方法を述べる。

この手法ではベクトルを合算して特徴を表すことで、ユーザの要求に沿いつつ意外性のある検索を行うことが期待できる。またユーザの要求、映画の特徴のそれぞれについてベクトルを合算するため、類似度を計算する回数が少なくなり計算量の減少が見込める。

具体的な手法について説明する。ユーザが好ましいと評価した特徴表現フレーズのベクトルをプラス、好ましくないと判断された特徴表現フレーズのベクトルをマイナスの値として合算することで、検索クエリとなるユーザの要求を表したベクトルを作成する。検索対象の映画は複数の特徴表現フレーズを持つが、ここではその映画のすべての特徴表現フレーズのベクトルを合算した1つのベクトルを映画の特徴量として利用する。そしてユーザの要求を表すベクトルと映画の特徴を表すベクトルの類似度を算出し、類似度の高い順にユーザに提示する。

次式(9)(10)はユーザの嗜好を表した特徴表現フレーズの集合である U と対象の映画が持つ特徴表現フレーズの集合 M の類似度を表す。 U の中にはユーザがこれまでに肯定的に評価したすべての特徴表現フレーズの集合である U_p と、これまでに否定的に評価したすべての特徴表現フレーズの集合である U_n が存在する。ここで \mathbf{u} はユーザが選んだ特徴表現フレーズのベクトルであり、ユーザが評価した数だけ存在する。 \mathbf{m} は映画 M が持つ特徴表現フレーズのベクトルであり、特徴表現フレーズ抽出時に設定した数だけ存在する。

$$S_{U,M} = \cos(\text{sum}_U, \sum_{\mathbf{m} \in M} \mathbf{m}) \quad (9)$$

$$\text{sum}_U = \sum_{\mathbf{u} \in U_p} \mathbf{u} - \sum_{\mathbf{u} \in U_n} \mathbf{u} \quad (10)$$

4.3 フィードバック用特徴表現フレーズの選出

1回目の検索クエリはユーザが検索し、出現した特徴表現フレーズを評価することで作成すると記したが、二回目以降の検索クエリの修正は検索結果をより正確にするために、システムが選んだ特徴表現フレーズをユーザに評価させることで行う。ここではフィードバック用にユーザに提示する特徴表現フレーズを選出する方法について述べる。

フィードバック用の特徴表現フレーズは、ユーザが一度特徴表現フレーズに評価を与え、検索結果が提示された状態で新たに提示される。ここから更にユーザの要求を絞り込むために、その時点ではユーザの要求に合致するか否か分からない特徴表現フレーズへのフィードバックが必要になる。そこで検索結果で上位に出現した映画群を持つ特徴表現フレーズのうち、ユーザが既に評価した特徴表現フレーズのいずれとも類似しない特徴表現フレーズをユーザに提示する。

この際の計算では word2vec による文ベクトルの \cos 類似度を利用し、設定した閾値以下の類似度を持つ特徴表現フレーズを選ぶという方法で行う。それによって得られた特徴表現フレーズをユーザに提示し、評価させる。このようにして評価した特徴表現フレーズを前回の評価集合に加えることで映画の検索を行う。

また三回目以降も同様にこれまでの特徴表現フレーズと類似しないものを選び、再提示することを繰り返す。

5 実験 1：特徴表現フレーズ抽出の評価

提案手法に基づいて実装した特徴表現フレーズ抽出システムを実行することによって、特徴表現フレーズの有効性を検証する。

5.1 実験概要

本実験では、抽出した特徴表現フレーズが映画の特徴を表しているかについて評価を行う。同様に、一つの映画から抽出した特徴表現フレーズ群が映画の主な特徴を網羅しているかについて評価を行う。特徴表現フレーズは、各映画ごとに各手法で 3 つずつ抽出し、それらが特徴を表しているかは、被験者の評価によって判断をする。評価項目は、その特徴表現フレーズが映画の特徴を表しているかどうかと、その特徴表現フレーズが映画に関連しているかである。回答は「はい」か「いいえ」の二択で行う。特徴を網羅しているかについては、まず被験者に映画の特徴を表すと思う単語をあげさせる。そして、全体の 3 分の 1 以上の被験者があげた単語をその映画の特徴的な単語とし、これらの単語が多く含まれているほど、特徴を網羅していると定義する。この際に、同じ意味を持つ単語は合わせて計上した。

被験者は 11 名で、事前の調査により視聴経験のある人数が多い映画を実験に用い、それぞれが観たことのある映画についてアンケートを行う。被験者には初めに、その映画の特徴を表す単語を 3 個から 10 個記述させる。その後、抽出した特徴表現フレーズを提示し、特徴表現と関連性の 2 項目について評価させる。

以下では、3.1 節で述べた手法を提案手法 1 とし、3.2 節で述べた品詞重み付きの手法を提案手法 2 とする。比較手法としては、文書要約アルゴリズム LexRank による要約を用いる。映画の全レビューを文書として LexRank によるスコアの高い文を選択し、これを提案手法と比較する。LexRank の類似度算出に用いるベクトルは、単語を次元とした TF-IDF モデルと、

表 1 提案システムのパラメータ

パラメータの種類	値
単語の最大出現映画数	15000
固有名詞の最低出現映画数	150
文の最大単語数	14
文の最低加算対象単語数	4
提案手法 2 での名詞の重み	1.0
提案手法 2 での動詞の重み	0.5
提案手法 2 での形容詞の重み	0.5
文の類似度の閾値	0.3

表 2 実験に用いた映画と被験者数

映画番号	被験者数	タイトル
映画 1	10	ハリリー・ポッターと賢者の石
映画 2	6	テルマエ・ロマエ
映画 3	9	ファインディング・ニモ
映画 4	9	サマーウォーズ

提案手法と同様の word2vec モデルを用いる。前者を比較手法 1、後者を比較手法 2 とする。

今回実装したシステムでの閾値や重みなどのパラメータは表 1 に記す。また、実験に用いた映画と、映画ごとの被験者の人数は表 2 に記す。

5.2 結果と考察

映画 1 の各手法で抽出した特徴表現フレーズと、被験者が「はい」と答えた比率を表 3 に記し、アンケートにより特徴的な単語とされた単語を下線で示す。同様に映画 2, 3, 4 についてはそれぞれ表 4, 5, 6 に記す。また、各手法での評価の平均と、単語を網羅している比率の平均を表 7 に記す。

表 7 より、提案手法 1 と 2 では提案手法 1 が特徴評価、関連評価共に値が高いことがわかる。また、提案手法と比較手法を比較すると、比較手法 1 の特徴評価が高く、提案手法の網羅性が高い。以下に詳しく考察する。

初めに提案手法 1 と提案手法 2 について比較する。品詞重みを用いない提案手法 1 に対して、提案手法 2 では品詞によってスコアに与える影響に重みを付けており、名詞の重みは動詞や形容詞の重みよりも高くなっている。これにより、映画 2 と 4 では提案手法 1 と 2 で異なる特徴表現フレーズが抽出できた。映画 2 を見ると、提案手法 2 では「阿部寛」や「原作」などの特徴量の大きい名詞と、それを説明する特徴量の小さい動詞の組み合わせによる特徴表現フレーズが抽出できており、品詞重みの効果が確認できる。映画 4 では、提案手法 2 にのみ出現した名詞として「時をかける少女」があり、この単語を含んだ特徴表現フレーズは評価が下がっている。しかし、映画の特徴を直接表してはいないものの、この例では同一の監督による日本のアニメ作品であるという共通点を持ち、この様な関係性のある特徴表現フレーズが抽出されることは、検索システムとしての用途を考えると望ましいものである。この様に、評価としては重みを使わない手法をやや下回ったものの、品詞重みの有効性を示す特徴表現フレーズを確認することができた。一方で、映画 1 と 3 では提案手法 1 と 2 で同じ特徴表現フレーズが抽

表 3 映画 1 の特徴表現フレーズとその評価 (特徴的な単語: 魔法, ファンタジー, 学校, 仲間)

手法	特徴評価	関連評価	特徴表現フレーズ
提案手法 1&2	0.4	0.9	ハリーとロンが、幼くて、かわいい!
提案手法 1&2	0.7	0.9	現実と魔法の世界の違いも面白かったなー。
提案手法 1&2	0	0.7	是非見るなら原作を読む前に見ましょう。
比較手法 1	0.1	0.6	最新作ではすっかり大人になってしまったハリーたち
比較手法 1	0.9	1.0	世界中にハリー旋風を巻き起こした第一作目です。
比較手法 1	0.4	0.9	ハリーとロンが、幼くて、かわいい!
比較手法 2	1.0	0.8	小さい子供が見ても楽しめるように、結構内容が盛だくさん。
比較手法 2	0.1	0.6	今と変わってるハリーが小さかったついに完結かあ
比較手法 2	0.2	0.6	実は今回初めてハリー・ポッターを見たのですが、すごい一言!

表 4 映画 2 の特徴表現フレーズとその評価 (特徴的な単語: 風呂, ローマ, 温泉, タイムスリップ)

手法	特徴評価	関連評価	特徴表現フレーズ
提案手法 1	1.0	1.0	ローマと日本をお風呂で結び付けるとは♪
提案手法 1	0.5	1.0	阿部寛の雰囲気と、演技と…笑えました。
提案手法 1	0.167	0.833	原作は読んだことありませんが、漫画であるらしいです。
提案手法 2	1.0	1.0	ローマと日本をお風呂で結び付けるとは♪
提案手法 2	0.5	1.0	なんと言っても、阿部寛演じるルシウスが、はまりすぎ。
提案手法 2	0.5	1.0	映画の前半は原作になぞらえてあり、笑えました。
比較手法 1	0.5	1.0	阿部ちゃんの演技、顔の表情がうまいし、面白い!
比較手法 1	0.167	1.0	阿部ちゃん以外のローマサイドの役者たち、北村一輝
比較手法 1	1.0	1.0	日本の風呂にタイムスリップした時の驚きの演技は爆笑。
比較手法 2	1.0	1.0	日本の風呂にタイムスリップした時の驚きの演技は爆笑。
比較手法 2	0.833	1.0	主要人物を日本人でキャストする大胆さに感動しました。
比較手法 2	0.5	0.833	クスッと笑ってホロリとなれる原作の完成度はかなり高いと思う。

表 5 映画 3 の特徴表現フレーズとその評価 (特徴的な単語: 魚, 海, 家族)

手法	特徴評価	関連評価	特徴表現フレーズ
提案手法 1&2	0.444	0.889	個人的には水槽の魚たちが気に入ってます。
提案手法 1&2	0.889	1.0	父と息子の成長に感動♪
提案手法 1&2	0.444	0.889	やはり家族で観るならディズニー×ピクサーは最適です。
比較手法 1	0.778	1.0	登場する魚の表情・行動・何気ない会話がめっちゃ可愛いです。
比較手法 1	0.444	0.889	個人的には水槽の魚たちが気に入ってます。
比較手法 1	0.556	0.444	親が無くとも子は育つ。
比較手法 2	0.222	0.444	私も子の親なので、すごく勉強になった。
比較手法 2	0.444	0.889	子供も大人も凄く引き込まれちゃいますよ。
比較手法 2	1.0	1.0	人間に捕らわれた魚の息子を探して、感動の再会でしょ!

出されたため、具体的な重みの数値については再考の余地がある。また、網羅性はどちらも同じ数値となり、品詞重みを用いることによる影響は見られなかった。

その他に、2つの提案手法に共通してみられる特徴として、3番目に抽出された特徴表現フレーズの評価が低いことがあげられる。提案手法では、既に選ばれた特徴表現フレーズとの類似性が高い候補がすべて除外されるため、適度に異なる特徴を表し、適度に映画の特徴を表す候補が除外されてしまうことがある。そのため、ここで用いる類似度の閾値は慎重に検討する必要がある。

次に提案手法と比較手法を比較する。比較手法 1 を見ると、提案手法とは異なる特徴表現フレーズがあがっており、提案手法よりも評価の平均値は高くなった。一方で、比較手法 2 を見ると映画 2 以外では他の手法に比べて低い評価となっている。

映画 2 の評価が全手法を通じて高かったことを考えると、特徴表現フレーズの抽出において word2vec によるベクトル化を用いた LexRank が適していないことがわかる。網羅性の観点で比較すると、提案手法は映画 1、映画 3 において比較手法より優れており、映画 2 においてはやや劣り、映画 4 においては比較手法 1 と同等であった。各手法の各映画の特徴を表す単語を網羅している比率を平均した値を比較すると、提案手法はどちらも比較手法よりも高い値であった。比較手法によって抽出した特徴表現フレーズを観てみると、映画 1 では「ハリー」、映画 2 では「阿部」など複数の特徴表現フレーズに同じ単語が見受けられる。LexRank の特性上、似通った内容の文がたくさん存在すると、それらの文同士でスコアを高め合ってしまう、似通った内容がスコア上位に並びやすい。そのため、網羅性の観点では提案手法が比較手法に勝ると考えた。

表 6 映画 4 の特徴表現フレーズとその評価（特徴的な単語：AI, 夏, 花札, 数学, 学生, インターネット）

手法	特徴評価	関連評価	特徴表現フレーズ
提案手法 1	1.0	1.0	田舎の大家族とネット仮想空間
提案手法 1	0.778	0.889	世界に誇れる日本のアニメ
提案手法 1	0.444	0.778	花札のときあれだけアカウントを
提案手法 2	0.889	0.889	家族は、この夏、世界を救った。
提案手法 2	0.778	0.778	花札のでこんなに盛り上がる映画は珍しい！
提案手法 2	0.111	0.667	時をかける少女もよかったが、心に残るものがある。
比較手法 1	1.0	1.0	田舎の大家族とネット仮想空間
比較手法 1	0.778	0.889	世界に誇れる日本のアニメ
比較手法 1	0.889	0.889	家族は、この夏、世界を救った。
比較手法 2	0.556	0.889	大人も子供も楽しめる、メッセージ性のある、すばらしい作品でした。
比較手法 2	0.111	0.667	アニメなのに身近に感じられて…もの凄く感動した！
比較手法 2	0.222	0.333	日本に生まれて日本の素晴らしいアニメが楽しめて…幸せ過ぎる。

表 7 各手法の評価値の平均

手法	特徴評価	関連評価	網羅性
提案手法 1	0.564	0.898	0.438
提案手法 2	0.554	0.884	0.438
比較手法 1	0.626	0.884	0.354
比較手法 2	0.516	0.755	0.208

表 8 部分ベクトル類似度による適合性フィードバックの評価

	検索要求	視聴欲求
初回	2.9	3.075
1 度目のフィードバック	2.5	2.85
2 度目のフィードバック	2.7	3.0

表 9 合算ベクトル類似度による適合性フィードバックの評価

	検索要求	視聴欲求
初回	2.85	3.0
1 度目のフィードバック	2.95	3.0
2 度目のフィードバック	2.925	3.05

今回の実験から、品詞重みが効果的に働いた特徴表現フレーズを確認することができた。また、網羅性の観点では、提案手法は比較手法よりも優れると判断した。今後は、品詞の重みや類似度の閾値を調整することによって抽出される特徴表現フレーズがどのように変化するのか、抽出する特徴表現フレーズの個数を変えることによって網羅性が向上するのかなどを考慮し、特徴表現フレーズ抽出の性能向上を目指す。

6 実験 2：検索システムの評価

6.1 実験概要

提案手法による検索システムにおける、特徴表現フレーズ選択による検索とフィードバックの効果を検証するために評価実験を行う。実験は被験者に検索システムを用いて映画を検索させ、検索結果を評価させることで行う。評価項目は 2 つで、「その映画は検索要求に合致しているか」と「その映画を観てみたいか」について、4 段階で評価させる。ここで、被験者はフィードバックを行わない初回の検索結果、1 度フィードバックを行った後の検索結果、2 度フィードバックを行った後の検索結果について評価をする。また、検索の終了後に、「特徴表現フレーズの選択によって自身の要求を表すことができたと思うか」についても 4 段階で評価させる。ここで用いる評価値は、「1：悪い」、「2：やや悪い」、「3：やや良い」、「4：良い」の 4 段階で、4 が最も良い評価となる。また、本実験は学生 8 人に対して行い、各手法について 1 回ずつ実験を行った。

実験で用いるシステムは以下の要件に従って実装した。

- 4.1 節の部分ベクトル類似度による検索と 4.2 節の合算ベクトル類似度による検索のそれぞれを用いたシステム 2 種類を実装した。

- 使用する特徴表現フレーズは品詞重みを用いたスコアリングで、固有名詞の重みを 1.0、一般名詞の重みを 0.8、動詞と形容詞の重みを 0.3 とし、1 件の映画に対して 3 件の特徴表現フレーズを抽出した。

- ユーザは、初めに単語を入力して特徴表現フレーズを検索し、出てきた特徴表現フレーズの中から自身の要求にもっとも近い 1 つを選択する。

- ユーザは、検索結果提示後に 10 個の特徴表現フレーズについてフィードバックを行い、そのそれぞれを検索条件に加えるかどうかを「はい」、「いいえ」、「どちらでもよい」の 3 択で入力する。

- 検索結果の映画は、一度の入力に対して 5 件出力する。

6.2 結果と考察

部分ベクトル類似度を用いた検索システムの実験結果を表 8 に、合算ベクトル類似度を用いた検索システムの実験結果を表 9 に記す。各項目の数値は、各回でのすべての評価値の平均値である。また、「特徴表現フレーズの選択によって自身の要求を表すことができたと思うか」という質問への回答の平均値は 3.0 であった。

2 度フィードバックを行った後での結果を見ると、合算ベクトル類似度による検索の方がスコアが高くなった。またすべての回の評価を平均した値もこちらの方が高い。合算ベクトル類似度による検索では初回の検索に対して、フィードバックを行

うと評価が上昇する傾向が見られた。一方で部分ベクトル類似度による検索では、初回に対して1度フィードバックを行うと評価が下がり、もう一度フィードバックを行うと評価がやや上昇するという結果になった。この類似度算出手法では、入力の特徴表現フレーズが一つの場合、その特徴表現フレーズを所持する映画が最上位の検索結果として出現するため、初回の評価値が高くなったと考えられる。また1度フィードバックを行った結果よりも、2度フィードバックを行った結果の方が評価が高くなったため、フィードバックの回数を更に増やすことでより検索要求に合った映画を提示できるのではないかと考えた。

この他に、個別の検索結果を検証することで、検索結果が低い評価となった原因について次のように考察した。

- フィードバックを繰り返すことによって、初めに選択したフレーズの特徴が薄れ、ユーザの思惑から外れていくことがある。

- ポジティブに評価したフレーズよりもネガティブに評価したフレーズが極端に多いと、一見して意味の分からない検索結果が提示される。

- システム内で特徴表現フレーズ中の一部の単語だけが強い意味を持ってしまい、ユーザの意図が正確に反映されないことがある。

これらの問題点に対して、適合性フィードバックとして一般的に用いられる Rocchio の式を元にフィードバックの重みを適切に扱える形式への変更、特徴表現フレーズのベクトル生成方法の見直しを行う。また、今回の実験では試行回数が少なかったこと、他手法との比較を行っていないことから、提案手法を適切に評価できなかったと考える。本手法の長所とする入力の簡便さ、特徴を組み合わせる映画を検索することの利点についての評価は今後の課題となるだろう。

7 ま と め

本稿ではレビューから映画の部分的な特徴を表現する短文を抽出する手法を提案し、この短文を特徴表現フレーズと名づけた。そして、特徴表現フレーズを用いた適合性フィードバックによる映画検索手法を提案した。

特徴表現フレーズの抽出では映画のレビューを形態素解析し、一文中の単語の TF-IDF 値を平均することで文のスコアを算出する手法を提案した。また、品詞ごとに異なる重みを与えることによってスコアリングする手法を提案した。スコアリングの後、cos 類似度によって類似性の低い文を選別し、特徴表現フレーズとした。

適合性フィードバックでは個々のベクトルの類似度を計算していく手法と、ベクトルを合算したのちに計算していく手法を提案した。いずれの手法もはじめにユーザが選んだ特徴表現フレーズを初期のベクトルとし、フィードバックによって要求に対する応答を正確にしていく。この際ユーザがフィードバックするのは、検索結果が上位の映画が持つ特徴表現フレーズのうち、これまでに評価した特徴表現フレーズとの類似性が低いものである。

実験1では、特徴表現フレーズの抽出を行った。また被験者にアンケートを行い、各特徴表現フレーズの評価を確認した。これによって品詞重みを用いた特徴表現フレーズ抽出の効果を検証、および提案手法が特徴の網羅の観点で優れていることを示した。実験2では、特徴表現フレーズ選択による検索と、フィードバックによる効果について検証をした。これによって、どちらも有効であるという結果が得られた。これらの結果を受けて、提案手法の有効性を確認することができた。また、システムや評価方法における問題点を発見できた。今後はこれらの調整を行った後に、より精密な評価が課題となる。

謝 辞

本研究の一部は、平成30年度科研費基盤研究(C)(課題番号:18K11551)によるものです。ここに記して謝意を表すものとします。

文 献

- [1] 岡田瑞穂, 藤井敦, “レビューテキスト間の類似度を用いた協調フィルタリング”, 言語処理学会 第18回年次大会 発表論文集, pp.711-714, 2012年
- [2] 梶見圭司, 北山大輔, “ユーザレビューを用いた全体的・部分的観点の類似に基づく映画推薦”, 第9回データ工学と情報マネジメントに関するフォーラム B5-4
- [3] 石田雄登, 打矢隆弘, 内匠逸, “内容に即した推薦文を提示する映画推薦システム”, “マルチメディア, 分散, 協調とモバイル(DICOMO2016)シンポジウム”, pp.1621-1627, 2016年
- [4] 林貴宏, 尾内理紀夫, “Web上のレビューを利用した映画推薦システム”, 人工知能学会論文誌, 30巻, 1号, SP1-E, pp.102-111, 2015年, 2017年
- [5] 幅野裕貴, 浦谷則好, “あらすじとレビューを用いた映画のキャッチコピー作成支援手法の提案”, 言語処理学会 第20回年次大会 発表論文集, pp.864-867, 2014年
- [6] 山下諒, 松下光範, “レビュー文を用いたコミックの内容判別手法の検討”, 2014年度人工知能学会全国大会, 4A1-4in, pp.1-4, 2014年
- [7] 久米雄介, 打矢隆弘, 内匠逸, “興味領域を考慮したTwitterユーザ推薦手法の提案と評価”, 情報処理学会研究報告, Vol.2015-ICS-179 No.1, pp.1-8, 2015年
- [8] 田中雄也, 田島敬史, “トピック空間上での文全体と各単語の類似度に基づく語重み付け手法”, 第10回データ工学と情報マネジメントに関するフォーラム D2-3, 2018年
- [9] 南大智, 牛尾剛聡, “書評SNSにおけるレビューの観点の違いを考慮したフィードバック型協調フィルタリング”, 第8回データ工学と情報マネジメントに関するフォーラム C7-2, 2016年
- [10] 杉山一成, 波多野賢治, 吉川正俊, 植村俊亮, “ユーザからの負担なく構築したプロファイルに基づく適応的web情報検索”, 電子情報通信学会論文誌 D-1 Vol.J87-D-1 No.11 pp.975-990, 2014年
- [11] Taku Kudo, Kaoru Yamamoto, Yuji Matsumoto, “Applying Conditional Random Fields to Japanese Morphological Analysis”, Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP-2004), pp.230-237, 2004.
- [12] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean, “Distributed Representations of Words and Phrase and their Compositionality”, Advances in Neural Information Processing Systems 26, pp.3111-3119, 2013