映画感想を記したblog記事内の肯定表現と否定表現に着目した映画推薦手法

Movie Recommendation Using Positive and Negative Expressions in Blog

今岡 千城 †
Tateki IMAOKA
imaoka@seman.cs.uec.ac.jp

林 貴宏 ‡
Takahiro HAYASHI
haya@seman.cs.uec.ac.jp

尾内 理紀夫 †‡
Rikio ONAI
onai@cs.uec.ac.jp

†電気通信大学大学院電気通信学研究科 情報工学専攻
Department of Computer Science, Graduate School of Electro-Communications,
The University of Electro-Communications
‡電気通信大学電気通信学部 情報工学科
Department of Computer Science, The University of Electro-Communications

概要

映画の感想が書かれた blog 記事には書き手の映画に関する嗜好が含まれていると考えることができる。あるユーザの blog 記事から映画に関する嗜好を抽出し、そのユーザと同じ嗜好を持つ他者が書いた映画の感想の blog 記事およびその記事中の映画を推薦することができれば、ユーザは嗜好に合った映画とその評判情報を得ることができる。そこで本研究では嗜好に合った映画の発見を支援するために、ユーザが記した映画の感想と他者が記した映画の感想を利用した映画推薦システムの開発を目指している。本稿ではそのシステムで用いる推薦手法について述べる。従来の推薦手法ではベクトル空間モデルにおけるtf-idfを利用した文書間類似度が用いられてきたが、ユーザの嗜好を抽出するためには単語の出現頻度だけではなく肯定表現や否定表現に着目する必要があると考える。そこで本稿では肯定表現と否定表現に着目した映画の blog 記事推薦手法を提案する。また提案手法を実装したシステムを試作した。

1. はじめに

家庭用ハードディスクレコーダの普及や、ストリーミング配信される映画の増加により、膨大な映画を見られる環境になりつつある。

このような環境では、自分の好みに合った映画を視聴できる機会が増える。自分の好みに合った映画を視聴したときの喜びは大きく、充実した時間を過ごしたと感じることができる。しかし、自分の好みに合わない映画を視聴してしまう機会も増加してしまう。自分の好みに合わない映画を視聴してしまったときの虚しさも大きく、無駄な時間を過ごしてしまったと感じる。

好みに合った映画を視聴する機会を増やし、好みに合わない映画を視聴してしまう機会を減らすために、自分の嗜好に合った映画を推薦するシステムが必要であると考える。

本研究では、自分の嗜好に合った映画を見つけるため、blog に着目した。現在、blog 上で自分の見た映画に関する感想を書くという行為が多く見られる。また、映画の感想が集められるウェブ

上のデータベースサイトに感想を多くの人が寄せている。それらの映画感想記事には書き手の映画に関する嗜好が含まれていると考えることができる。あるユーザの映画感想記事から映画に関する嗜好を抽出し、そのユーザと同じ嗜好を持つと考えられる他者が書いた映画感想記事およびその記事中の映画を推薦することができれば、ユーザは嗜好に合った映画とその評判情報を得ることができると考えた。

そこで本研究では嗜好に合った映画の発見を支援するために、ユーザが記した映画の感想(blog 記事)と、他者が記した映画の感想(blog 記事)を利用した映画推薦システムの開発を目指している。このシステムにユーザがblog記事を入力すると、システムはユーザの嗜好をその blog 記事から抽出すると共に、ウェブ上の blog 記事の中から、ユーザと同じ嗜好を持つ他者の blog 記事を探し、ユーザに提示する。

以下、2 章では従来用いられてきた推薦手法とその問題について述べる。3 章では本研究で提案する推薦手法について述べる。4 章で提案した推薦手法を実装した試作システムについて述べる。5 章では提案した推薦手法に対する評価実験について述べる。

2. 従来手法

ユーザの書いた文書をもとにして、他の文書を推薦するシステムでは、文書間類似度に基づく手法が利用されることがある。ユーザの書いた文書から特徴語を抽出し、同じ特徴語が使用される文書ほど類似度が高いと判断する。この類似度を推薦スコアとし、推薦スコアの高い文書をユーザに提示する。特徴語として tf-idf 値が高い単語を選択する手法が一般に利用される。そして、特徴語を用いた文書間の類似度にはベクトル空間モデルが用いられる。各文書を特徴語の tf-idf の値を要素とするキーワードベクトルで表現し、そのコサイン値を類似度すなわち推薦スコアとする。この手法における推薦スコア T を式(1)〜(4)で示す。ただし、ユーザ文書 d における特徴語の集合を $W_d=\{w_1,w_2,\cdots,w_N\}$、文書 d'の特徴語の集合を $W_{d'}=\{w'_1,w'_2,\cdots,w'_M\}$、tfidf(w)を単語 w の if-idf 値とする。

$$T = \frac{\langle V_d \cdot V_{d'} \rangle}{|V_d||V_{d'}|} \quad (1)$$

$$V_d = \left(f(w_1), f(w_2), \ldots, f(w_N)\right) \quad (2)$$

$$V_{d'} = \left(f(w'_1), f(w'_2), \ldots, f(w'_M)\right) \quad (3)$$

$$f(w) = tfidf\ (w) \quad (4)$$

この手法は、ユーザの書いた文書に類似した文書はユーザの興味を反映しているという仮定に基づいている。

しかしこの方法では単語の出現頻度のみが注目されているため、例えば「この映画の音楽は良い」と「この映画の音楽は良くない」という反対の嗜好情報を持つ二文を区別することができない。blog 記事の書き手の嗜好を抽出するためには単語の出現頻度のみではなく、「面白い」「面白くない」「良い」「悪い」といった肯定的な評価表現や否定的な評価表現に着目する必要があると考える。

3. 提案手法

本章では blog 記事中の肯定的な評価表現(以下、肯定表現)と否定的な評価表現(以下、否定表現)に着目した映画 blog 記事の推薦手法を提案する。

以下、3.1 節で肯定表現と否定表現を考慮した

特徴語である興味語について述べる。3.2節で具体的な処理手順を述べる。

3.1 興味語

本研究ではユーザが書いた blog 記事中で肯定表現(例えば「良い」「面白い」)の周辺に出現する単語群がユーザの嗜好を表すと仮定する。これらの単語を興味語と定義する。そこで、他者の blog 記事中でこれらの興味語が存在し、その周辺に肯定表現が出現するとき、その blog 記事はユーザと同じ嗜好を持つ他者が書いたと判断する。例えばユーザが blog 記事中に「脇役の演技がとてもよかった。」と書いた場合「脇役」「演技」が興味語となる。そして他者の blog 記事中で「脇役の俳優による好演が光った。」「出演者の演技が非常に良かった。」などと書かれているとき、その blog 記事はユーザと同じ「脇役」「演技」という興味語を持つ他者が書いたものと判定する。

また、否定表現(例えば「良くない」「つまらない」)の周辺に出現する単語群もユーザの嗜好を表すと仮定し、同様に興味語と定義する。そこで、他者の blog 記事中でこれらの興味語が存在し、その周辺に肯定表現が出現する時、その blog 記事はユーザと同じ興味語を持つ他者が書いたと判断する。例えば「音楽が良くなかった。」という文は音楽に興味を持つが故に音楽が期待はずれでがっかりしていることを表現していると考えられるので、「音楽が良かった。」という文を書いた他者はユーザと同じ嗜好を持つと判断する。

3.2 処理手順

上記の仮定に基づいた2種の推薦手法の具体的な処理手順を以下に述べる。

(手順1) ユーザの blog 記事 d の肯定表現と否定表現を検出する。

(手順2) blog 記事 d の PKV,NKV を求める。

[PKV] 肯定表現が含まれる文に現れる名詞を興味語とし、それらの集合を W_p とする。それらの tf-idf 値を要素とする正のキーワードベクトル PKV を求める。

[NKV] 否定表現が含まれる文に現れる名詞を興味語とし、それらの集合を W_n とする。それらの tf-idf 値を要素とする負のキーワードベクトル NKV を求める。

記事 d に出現する名詞の集合を $W=\{w_1,w_2,\cdots,w_N\}$ とし、式(5)~(8)に PKV,NKV の算出法を示す。

$$PKV=\left(f_p(w_1),f_p(w_2),\ldots,f_p(w_N)\right) \quad (5)$$

$$NKV=\left(f_n(w_1),f_n(w_2),\ldots,f_n(w_N)\right) \quad (6)$$

$$f_p(w)=\begin{cases} tfidf(w) & (if\ w\in W_p) \\ 0 & (if\ w\notin W_p) \end{cases} \quad (7)$$

$$f_n(w)=\begin{cases} tfidf(w) & (if\ w\in W_n) \\ 0 & (if\ w\notin W_n) \end{cases} \quad (8)$$

(手順3) 二種の推薦スコアを算出する。

[推薦手法 1]ユーザの記事 d における PKV_d と、他者の blog 記事 d'における $PKV_{d'}$ のコサイン値 $S_1(d,d')$ を算出し、推薦スコアとする。

[推薦手法 2]ユーザの blog 記事 d における NKV_d と、他者の blog 記事 d'における $PKV_{d'}$ のコサイン値 $S_2(d,d')$ を算出し、推薦スコアとする。

$S_1(d,d'),S_2(d,d')$ を式(9)(10)に示す。

$$S_1(d,d') = \frac{\langle PKV_d \cdot PKV_{d'} \rangle}{|PKV_d||PKV_{d'}|} \quad (9)$$

$$S_2(d,d') = \frac{\langle NKV_d \cdot PKV_{d'} \rangle}{|NKV_d||PKV_{d'}|} \quad (10)$$

(手順4) 各推薦手法に対し、推薦スコアが大きい記事をユーザに提示する。

4. 試作システム

提案手法を利用した映画推薦システムを試作した。システムの概要を図 1 に示す。

前処理部はウェブ上の blog 記事を収集し、本文の抽出や評価表現、興味語の抽出などの前処理を行う。推薦処理部はユーザがシステムに blog 記事を投稿すると、その記事に対し推薦処理を行い、推薦記事を決定し、ユーザに提示する。

以下 4.1 節で前処理部について述べる。4.2 節で推薦処理部について述べる。4.3 節で PKV,NKV を求める方法の詳細について述べる。

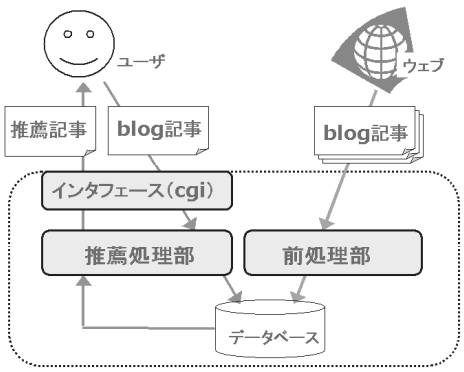

図 1 システム概要

4.1 前処理部

前処理部は以下の手順で実行される。

(手順1) 映画の感想の書かれたウェブページ (html ファイル)を収集する。本システムで収集するページは、ブログポータルサイトにおいて映画カテゴリに登録されている blog 記事ページ、および映画の感想が多く投稿されている映画データベースサイトのページである。

(手順2) 収集したページから記事の URL、記事のタイトル、本文を抽出する。本システムが収集するページは定型的な html のタグ構造を持つため、正規表現を用いて各データを抽出する。また、映画データベースサイトの記事については、映画のタイトルについても抽出する。

(手順3) 記事の本文に対し形態素解析を行い、単語単位に分解する。本システムでは MeCab[1]を用いる。

(手順4) 記事の URL、記事のタイトル、記事本文、抽出された単語の集合をデータベースに格納する。本システムではデータベースに MySQL[2]を用いる。

4.2 推薦処理部

推薦処理部は以下の手順で実行される。

(手順1) ユーザが投稿した記事の本文に対し形態素解析を行い、単語単位に分割する。

(手順2) ユーザが投稿した記事の PKV,NKV を求める。4.3 節で詳しく述べる。

[1] http://mecab.sorceforge.jp
[2] http://www.mysql.com

(手順3) データベースから他者の記事を取り出す。これを推薦候補記事とする。

(手順4) 推薦候補記事の PKV を求める。

(手順5) ユーザ記事の PKV と推薦候補記事の PKV を用いて、3.2 節における式(9)で示した推薦手法 1 の推薦スコア S_1 を求める。またユーザ記事の NKV と推薦候補記事の PKV を用いて、3.2 節における式(10)で示した推薦手法 2 の推薦スコア S_2 を求める。

(手順6) 推薦手法 1 における推薦スコア S_1 が大きい推薦候補記事上位 10 件、および推薦手法 2 における推薦スコア S_2 の大きい推薦候補記事上位 10 件を推薦記事として、スコア順にユーザに提示する。

4.3 PKV,NKV の作成

1 つの記事に対し PKV,NKV を計算するため、まず記事の本文を文単位へ分割する。文は句点、ピリオド、クエスチョンマーク、エクスクラメーションマークが文末に現れると仮定し、これらの記号の出現位置で本文を分割し、分割されたそれぞれの区間を文とした。

次に評価表現が現れる文を選出する。評価表現として評価値表現辞書の中の肯定表現、否定表現を使用した。肯定表現と否定表現の判断は主観で行った。

肯定表現が現れた文に含まれる名詞の集合 W_p、否定表現が現れた文に含まれる名詞の集合を W_n、その記事に出現する全ての名詞の集合を W として、3.2 節における式(5)〜(8)で定義した PKV,NKV を算出する。

5. 評価実験

5.1 実験 1

5.1.1 目的と方法

実験 1 では、提案した 2 種の推薦手法と 2 章で述べた従来手法の推薦精度を評価する。推薦した記事のうち、ユーザがその記事中の映画を見たいと思った記事の割合によって評価した。

17 人の被験者に映画の感想記事を書いてもらい、推薦手法 1、推薦手法 2、従来手法を用いて選出された推薦記事を、それぞれ最大 10 件被験者に提示し、それら推薦記事をアンケートにより評価してもらった。

推薦された映画がユーザにとって未知である場合の推薦記事に対する評価を得るために、まず以下の質問に解答してもらった。

質問1:推薦記事中の映画は見たことがありますか？(はい・いいえ)

質問 2:推薦記事中の映画のタイトルは知っていましたか？(はい・いいえ)

質問1に「いいえ」、質問2に「いいえ」と回答された記事を評価対象記事とした。続いて評価対象記事に関して、以下の質問を行った。

質問3:記事を読んでその映画を見たいと思いましたか？5 段階で評価してください。(強く思う・やや思う・どちらとも思わない・あまり思わない・全く思わない)

各推薦手法の精度を測るための尺度 U を式(11)で定義する。

$$U = \frac{X}{Y} \qquad (11)$$

Y は推薦記事数、X は推薦記事を読んで記事中の映画をユーザが見たいと感じた記事の数すなわち、質問 3 において「強く思う」もしくは「やや思う」と回答された記事の数である。

5.1.2 結果と考察

アンケートの結果から各推薦手法の推薦精度 U を計算した。

推薦手法 1 の場合、Y=225、X=75 で U=0.33 であった。また、推薦手法 2 の場合 Y=159、X=51 で U=0.32 であった。つまりどちらの手法の場合も約 32%の割合でユーザがその映画を見たいと解答する記事を推薦できたと言える。それに対して従来手法の場合、Y=233、X=65 で、U=0.28 であった。つまり従来手法では約 28%の割合でユーザがその映画を見たいと解答する記事を推薦できたと言える。(表 1)

表 1 推薦精度

推薦手法	推薦精度(%)
推薦手法 1	33
推薦手法 2	32
従来手法	28

以上から、提案手法と従来手法の推薦精度の差は 5%であった。提案手法の精度が従来手法に比べ高い精度であったが、大きな差は見られなかった。これは、提案手法における興味語と従来手法における特徴語が類似していることが原因であると推測される。事実、従来手法によって提示された記事(223 件)のうち、ユーザ記事と推薦記事に共通して含まれる単語の少なくとも1つが、推薦候補記事において PKV に含まれている記事(119)は約 50%あった。つまり従来手法で推薦された記事のうち約半数の記事で、提案手法におけるスコア計算に同じ単語を用いていると言える。

興味語と従来手法における特徴語の関係の検討は今後の課題である。

5.2 実験 2

5.2.1 目的と方法

本研究では、ユーザの興味語が他者の記事中で肯定表現の周辺に現れるとき、ユーザはその映画に対する興味を喚起されるという前提のもと、手法を提案した。そこで、実験 2 ではユーザが興味を喚起された文には肯定表現が含まれていることを確認する。

実験 1 におけるアンケートに続いて、以下のアンケートを行った。

質問 4 :記事内のどの文を読んで質問 3 の回答にいたりましたか？文を選んでください。(複数回答可)

質問 4 の回答は、記事の文の先頭にチェックボックスとつけて提示し、チェックボックスにチェックを入れることで解答してもらった(図 2)。

質問 3 において「強く思う」または「やや思う」と回答された記事は、ユーザがその映画に対する興味を喚起された記事と言うことができる。そしてユーザがその映画に対する興味を喚起された記事中の文のうち、質問 4 でチェックボックスにチェックが入れられた文は、興味を喚起されるきっかけとなった文と言える。そこで、ユーザが興味を喚起されるきっかけとなった文のうち、肯定表現

が含まれる文の割合を調べた。また比較対象として、ユーザが興味を喚起された記事の中でユーザが興味を喚起されたきっかけとして選ばなかった文(質問 4 の回答でチェックボックスにチェックを入れられなかった文)のうち、肯定表現を含む文の割合を調べた。

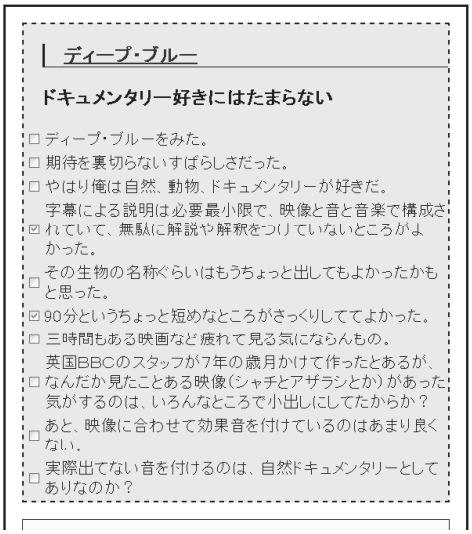

図 2 質問 4 の回答欄

5.2.2 結果と考察

実験 1、実験 2 のアンケートから得られた結果を表 2 に示す。

ユーザが興味を強く喚起された記事(質問 3 において「強く思う」と回答された記事)の文中、ユーザが興味を喚起されるきっかけとなった文(質問 4 において選択された文)は 137 文であった。そのうち 66 文において肯定表現が含まれていた。

また、ユーザが興味を喚起された記事(質問 3 において「強く思う」または「やや思う」と回答された記事)の文のうち、ユーザが興味を喚起されるきっかけとなった文(質問 4 において選択された文)は 362 文であった。そのうち 179 文が肯定表現を含んでいた。つまり、ユーザが興味を喚起されるきっかけとなった文の約 49% (66/137=0.48,179/362=0.49) は肯定表現を含んでいた。それに対し、ユーザが興味を喚起されるきっかけとならなかった文のうち評価表現を含む割合は約 31%であった。

つまりユーザが興味を喚起される文には高い割合で肯定表現が含まれているので、肯定表現の周辺に興味語が現れる他者の記事を推薦する手法は妥当であると言える。

表 2 質問 3 と質問 4 の回答

質問 4 の回答 ＼ 質問 3 の回答		強く思う	やや思う	どちらとも思わない	あまり思わない	全く思わない	合計
選択された文	肯定表現を含む文	66 文	113 文	56 文	91 文	77 文	403 文
	肯定表現を含まない文	71 文	112 文	102 文	176 文	165 文	626 文
選択されなかった文	肯定表現を含む文	51 文	135 文	59 文	182 文	123 文	550 文
	肯定表現を含まない文	149 文	252 文	183 文	409 文	257 文	1250 文

6. 関連研究

MineBlog[1]は、blog を書いているユーザに対し、新たな興味発見を支援することを目的としたシステムである。本研究と同様、ユーザの書いた blog 記事からユーザの嗜好を抽出し、他者の blog 記事を推薦するシステムである。MineBlog では、関連性、相違性、話題性の 3 つの尺度で推薦候補記事を評価し、推薦記事を決定する。このうち、関連性は推薦候補記事がユーザの書いた blog 記事とどの程度関連する話題を含んでいるかを測る尺度で、ユーザの書いた blog 記事の特徴語(tf-idf 値の大きい単語)と推薦候補記事の特徴語によって決定される。この手法は特徴語の選出に tf-idf 値の大きさのみを用いているという点で、2 章で述べた従来手法と類似している。本研究とは、tf-idf 値の大きさに加え周辺に出現する否定表現、肯定表現を用いているという点で異なっている。

阿部ら[2][3]は、柔軟で多角的な映画の探索を行うことを可能にするため、映画のコメントを用いて映画を分類する研究を行っている。本研究と同様、嗜好に合った映画を発見することを支援することが目的である。[2]では、2 章で述べたような tf-idf 値とベクトル空間モデルを用いた文書間類似度を計算し、映画を分類を試みている。[3]では、ナイーブベイズ分類を用いて映画を分類している。この研究では、多くの人によるコメントの集合から映画の特徴を抽出し、分類に利用しているが、本研究ではコメントから得られた映画の特徴ではなく、感想記事から個人の嗜好を抽出し利用するという点で異なっている。また、この研究では肯定表現、否定表現に着目していない。

7. おわりに

blog 記事中の肯定表現と否定表現に着目した映画推薦手法を提案し、試作システムを実装、評価した。そして以下の 2 点を確認した。

- 提案手法は従来手法より5%ほど高い割合でユーザに見たいと思わせる映画を推薦できた。
- ユーザが映画に対する興味を喚起される文には肯定表現が高い割合で含まれる。

今後の課題として、5.2.2 節の結果と考察で述べた、提案手法における興味語と従来手法における特徴語の関係の検討がある。

また映画分野のドメイン知識利用を検討する必要がある。例えば映画においてユーザが注目する点は出演者名やジャンル等に多いと考えられる。これらを表す単語(人名やアクション、ホラー等のジャンル名)を積極的に興味語として利用することで、より映画に関する嗜好として適した興味語の抽出ができるのではないかと考えている。

参考文献

[1]森本和伸,林貴宏,尾内理紀夫,"MineBlog:興味発見を支援する blog 推薦システム",情報処理学会論文誌,Vol.47 No.4 pp.1171-1180 (2006)

[2]阿部倫子,細野公男,中川裕志,"コメント文を利用する映画ナビゲーション", 言語処理学会第7回年次大会 (2001)

[3]阿部倫子,田中久美子,中川裕志,"コメントを用いた映画の分類",情報処理学会 NL 研究会 NL-150,pp.105-110 (2002)