

E-015

## 既読率に基づいた記事推薦システムについての研究

Article recommendation system based on ratio of partial contents already finished reading among the whole

原田 真熙† 和田雄次‡  
Masaki Harada Yuji Wada

## 1. はじめに

## 1.1 背景

近年, ニュース記事やまとめサイト, ブログ等も含め, WEB 上にはユーザー投稿型のサイトが多数ある. そういった現在の WEB 上にある記事投稿サイトで使われている推薦方法の一つに, ユーザーの閲覧した記事を基に, それに類似した記事を推薦するというものがある.

しかし, 閲覧した記事の中には閲覧したものの途中までしか読んでない記事が存在するなど, 内容を実際に読んでみなければユーザーがその記事にどの程度興味を寄せているかが分からない. そのような, 閲覧したが内容に興味がなかった記事を読み終えた記事と同じように扱ってしまうと, 興味のない記事を推薦する可能性がある.

そこで本研究では, ユーザーが閲覧した記事をどの程度まで読んでいくかという情報を”既読率”とし, これを記事同士の類似度と掛け合わせることでユーザーの興味が高い記事を優先して推薦するシステムの構築を目的とする.

## 1.2 関連研究

文書をベクトル化することで特定の記事と類似している記事を推薦する研究[1]や, レビュー文をベクトル化することで, 映画同士の類似度を算出し推薦する研究がされている[2]. しかし, 本研究の既読率のような値を組み合わせることで, ユーザーの興味の深さを考慮する研究は行われていない.

## 2 既読率に基づいた記事推薦システム

## 2.1 推薦の流れ

本研究で提案する推薦システムの流れを図 1 に示す.<sup>1</sup>

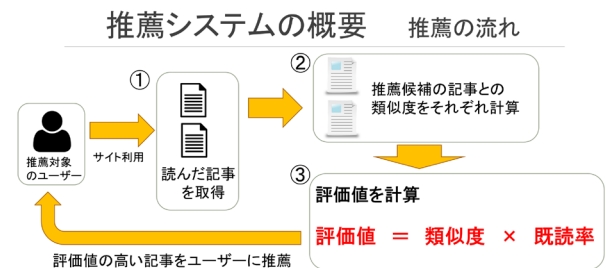


図 1 提案する記事推薦システムの流れ

サイトを利用したユーザーの読んだ記事を取得し, 推薦候補の記事との類似度をそれぞれ計算する. その後, 類似度に既読率を掛け合わせてそれを”評価値”とする.

$$\text{評価値} = \text{類似度} \times \text{既読率} \dots(1)$$

評価値が高い記事をユーザーに推薦する.

## 2.2 既読率

1つの文書をどの程度まで読み進めたか, というものを表す数値として扱う. 具体的には, web サイトの記事ページ全体の高さをスクロール量で割ることで算出する. また, 本研究ではどこまで読み進めたかを仮定して計算する.

## 2.3 記事同士の類似度

記事同士の類似度計算には, ベクトルの内積を利用して類似度を算出する方法であるコサイン類似度を用いる[3]. 記事をベクトル化する手法として Doc2Vec を利用する. Doc2Vec は, 任意の長さの文章をベクトル化する Paragraph Vector の一つである[4]. Doc2Vec を用いて記事をベクトル化することで, コサイン類似度を用いた類似度計算を行う.

## 3 評価実験

## 3.1 データセット

サンプルデータとして Livedoor ニュースコーパスを用いた. NHN Japan 株式会社が運営する「livedoor ニュース」のうち, クリエイティブ・コモンズライセンスが適用されるニュース記事を収集し, 可能な限り HTML タグを取り除いて

†東京電機大学大学院情報環境学研究所  
‡東京電機大学システムデザイン工学部

作成されたものである[5]。Livedoor ニュースコーパスには7,363件のニュース記事が収集されている。本研究ではこれをサンプルデータとして用いた。

### 3.2 実験内容

本研究では、サンプルデータを Doc2Vec を用いてベクトル化し、それぞれの記事を文書 ID1, 2, 3...とする。あるユーザーが文書 ID1 の記事を6割(既読率 0.6)読み、文書 ID2 の記事を全て読んだ(既読率 1.0)と仮定し、文書 ID1, 文書 ID2 の記事と類似した記事の上位 5 件と、式(1)を用いて評価値を算出した結果を比較する。

実験結果を表1に示す。

表1 文書 ID1 との類似度と評価値

文書ID1 (既読率0.6)	文書ID1との 類似度	評価値
文書ID5345	0.212	0.127
文書ID3681	0.207	0.124
文書ID5665	0.189	0.113
文書ID4917	0.185	0.111
文書ID5136	0.185	0.111

表2 文書 ID2 との類似度と評価値

文書 ID2 (既読率 1.0)	文書 ID2 との 類似度	評価値
文書 ID5512	0.194	0.194
文書 ID6083	0.191	0.191
文書 ID3975	0.190	0.190
文書 ID6000	0.183	0.183
文書 ID6595	01.83	0.183

### 4 考察

類似度の近い記事が並んだ場合、既読率の高い記事の方が最終的な評価値としては高くなっているという結果がみられるため、サンプルデータ上では、より多く読み進めた記事に類似している記事を優先して推薦するという手法が有効である場合を示すことが出来た。しかし、既読率による影響が大きく、既読率が低い記事に対して類似度が高い記事はほとんど推薦することが出来ない。評価値の計算を行う

際に改善の余地があると考えられる。

### 5 まとめ

本研究では、記事の類似度と既読率を積算することで、ユーザーの興味が高い記事を優先して推薦する手法を提案した。

提案した手法では Doc2Vec を用いて記事をベクトル化し、類似度計算を行う。そして、算出した数値に記事の既読率を掛け合わせてそれを評価値とする。実験では、livedoor ニュースコーパスの記事をサンプルデータとして用いて実験を行った。類似度のみで見た場合と既読率を考慮した場合を比較することで、有効な場合があることを確認した。

今後は GUI を組みユーザーテストを行うことで提案手法の効果を検証する。そのフィードバックを得た上で、提案手法をユーザーに有効な形で用いるために、類似度や評価値の計算、手法の改善を行う予定である。

### 6 参考文献

- [1] 山本祐生, 三川健太, 後藤正幸, "トピックモデルに基づく協調フィルタリングによる文書推薦手法について", [https://www.ieice.org/publications/search/bin/pdf/link.php?fname=E-015.pdf&year=2016&ConfCd=F&id=FIT0000011951&lang=J&tbl=conf\\_fit](https://www.ieice.org/publications/search/bin/pdf/link.php?fname=E-015.pdf&year=2016&ConfCd=F&id=FIT0000011951&lang=J&tbl=conf_fit), (最終閲覧日: 2020年1月9日)。
- [2] 榎見 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", <https://db-event.jpn.org/deim2017/papers/85.pdf>, (最終閲覧日: 2020年1月10日)。
- [3] Dietmar Jannach, Markus Zanker, Alexander Felfernig, Gerhard Friedrich 著, 田中克己・角谷和俊訳, 『情報推薦システム入門 理論と実践』, 共立出版株式会社, 2013。
- [4] Le, Q. and Mikolov, T., "Distributed Representations of Sentences and Documents", CoRR, abs/1405.4053, pp. 1-9, 2014.
- [5] 株式会社ロンウィット, <https://www.rondhuit.com/download.html#ldcc>