

図書推薦システムの開発

図書館班: 小野碧瑠 鈴木結愛佳 深堀琴子

(1) 研究の背景

◎福島県立図書館の来館者数が東日本大震災後に減っているため、もっと利用しやすい図書館を目指す。

→新たなシステムの導入が有効

推薦文をもとにして検索することで、利用者に寄り添った図書推薦システムを作れると考えた。

推薦文は福島高校の図書委員、教員の皆さんが執筆されたものをお借りした。

◎まずは精度の高い検索機能完成を目指す。

(2) 実験に使用したアルゴリズム

集合を用いた方法を行う。また、重要度を出す計算にはIDF値を計算する方法をもとにして自作した式を用いた。

言語はPython3を使う。

- ①利用者が打ち込んだ文章を形態素解析エンジンMeCabを用いて品詞分解する
- ②品詞分解した単語の中から、重要度(※)が高い単語をいくつか取り出す
- ③それらの単語について、その単語が含まれている推薦文の集合をそれぞれ作る
- ④その集合が最も重なる推薦文を選び出し、その推薦文を持つ本を推薦する

(※)重要度 = $-(5/\text{すべての推薦文数}) * \text{その単語を含む推薦文数} + 6$



(3) 実験内容と実験方法

作成したプログラムによって推薦した結果と、人が推薦した結果がどれだけ近いかを測定し、人の推薦結果とプログラムの推薦結果が一致することを目指す。

<実験内容>

実験1: 上記のアルゴリズムをそのまま試す

実験2: 上記③において、入力された文の活用している単語を終止形に直してから集合を作成する

<実験方法>

(1)推薦文をもとに、利用者の検索文の例とそれに対して人が推薦した結果のセットをいくつかを作成

例)文章⑥「私は部活動に取り組む物語が読みたいです」⇒「響け!ユーフォニアム~北宇治高校吹奏楽へようこそ!~」を人が推薦

(2)作成したプログラムに検索文を入力し、その結果と人が推薦した結果を比較する。

例)文章⑥「私は部活動に取り組む物語が読みたいです」と入力⇒「松永真、デザインの話。」が推薦され、出力された。

(3)人が推薦した本と同じ本がある場合には1冊につき1点を「一致点数」に、人が推薦した本とは違う誤った本を推薦した場合には1冊につき-1点を「余分点数」に加算して、一致点数と余分点数の合計値を「点数」とすることで近さを点数化する。

例)文章⑥だと、人が推薦した本と一致しなかったので0点加算(一致点数)、人が推薦しておらず関係ない本を1冊推薦したため-1点加算(余分点数)する。一致点数と余分点数の合計は-1点であるため、文章⑥の点数は-1点である。

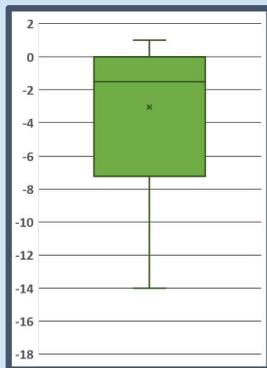
以上の操作を18種類の入力文で行った。

(4) 実験結果

表中の番号は文章の番号をさす。点数を箱ひげ図にあらわした。

<実験1> 結果は以下の通り

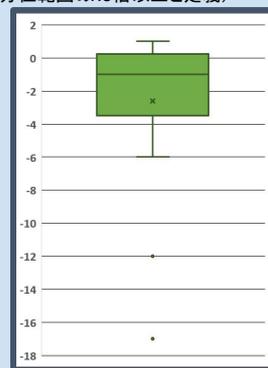
	点数	満点	一致	余分
①	0	2	1	1
②	-2	2	0	2
③	-1	2	1	2
④	-2	1	1	3
⑤	-1	1	0	1
⑥	-1	1	0	1
⑦	1	2	1	0
⑧	0	2	1	1
⑨	1	2	1	0
⑩	-2	2	1	3
⑪	0	1	1	1
⑫	-9	1	1	10
⑬	-3	1	1	4
⑭	-7	2	1	8
⑮	-8	1	0	8
⑯	-8	1	0	8
⑰	1	1	1	0
⑱	-14	2	0	14



◎全体的に負の点数が多く、満点とかけ離れているものもあった。

<実験2> 結果は以下の通り(外れ値は四分位範囲の1.5倍以上と定義)

	点数	満点	一致	余分
①	0	2	1	1
②	-12	2	0	12
③	-2	2	2	4
④	1	1	1	0
⑤	-1	1	0	1
⑥	-5	1	0	5
⑦	0	2	0	0
⑧	-1	2	0	1
⑨	1	2	1	0
⑩	0	2	1	1
⑪	1	1	1	0
⑫	-1	1	0	1
⑬	-1	1	1	2
⑭	-6	2	1	7
⑮	-3	1	0	3
⑯	-17	1	0	17
⑰	1	1	1	0
⑱	-2	2	1	3



◎<実験1>よりも範囲が狭まったが、外れ値が出た。

(5) 実験の考察

◎実験2において実験1より精度が大幅に下がってしまった例があった。

→終止形に戻す際に「ある」が漢字に直されてしまい、推薦文にはあまり含まれない「有る」という字になる事例により、本来低くなると想定される単語の重要度が上昇した。

◎実験1と実験2の箱ひげ図を比較

→実験2の方が「有る」によって出た外れ値を除けば範囲が狭く、

実験1よりも精度が上がったのではないかと評価できる。

→実験1と比べ、誤った文章を推薦する事例が減少した。

(6) 今後の展望

・活用語を終止形に直した際に、変化した漢字が重要度に影響を及ぼさないような方法を模索する。

・プログラムで参照する推薦文に関して、ほかの文章(本についている説明文等)にしたときにどう正確性が変化するかを調べる。

参考文献・謝辞

・【Python】MeCab(形態素解析器)を使用して文章をカテゴリ分類する

<https://zerofromlight.com/blogs/detail/56/#parse>

・推薦文を執筆してくださった皆様に感謝いたします。