オンライン小説の流行語抽出

堺, 雄之介
九州大学工学部電気情報工学科

伊東, 栄典
九州大学情報基盤研究開発センター

出版情報：情報処理学会全国大会講演論文集. 82, pp.6T-01-, 2020-03-06. 情報処理学会
バージョン：

オンライン小説の流行語抽出

堺 雄之介 *　伊東 栄典 **
(九州大学 *工学部電気情報工学科 **情報基盤研究開発センター)
* y.sakai.a96@s.kyushu-u.ac.jp, ** ito.eisuke.523@m.kyushu-u.ac.jp

1　はじめに

大衆の動向が把握できれば商機につながるため，Twitterからの流行語抽出や，Google トレンドでの検索語トレンド分析が行われている．近年オンライン小説が人気である.「小説家になろう」,「カクヨム」等のサイトは多くの利用者が小説を読み，また作者も小説をサイトに登録するようになっている．本研究では「小説家になろう」の小説メタデータを集めた．メタデータには題名・作者・あらすじ・キーワード等が含まれている．このメタデータ群を対象に，分野ごとかつ月ごとの流行語分析を行う．流行語分析では，簡単な単語出現頻度による分析とともに，単語の分散表現による類似語抽出からの類似単語集約によるトレンド分析も行う．また流行語分析ツールも作成した．

2　なろう小説 API

なろう小説 API [1] は「小説家になろう」に掲載されている小説メタデータを取得できる API である．この API の出力として小説名やあらすじなど，計 40 項目のデータが得られる．その中で本研究にて使用したデータ項目を表 1 に示す．本研究では 2004 年 4 月 20 日から 2019 年 11 月 15 日の期間に投稿された 693,304 件の小説のメタデータを用いた．

表 1: 使用する小説メタデータ

要素	説明
title	小説名
ncode	N コード
story	小説のあらすじ
keyword	キーワード
general_firstup	初回掲載日

3　データ処理と単語の分散表現取得

本研究では流行語抽出のために 2 つの方法を適用する．1 つ目は単語の出現頻度を数え上げる方法である．2 つ目は単語の出現頻度に加え，その単語に関連する単語の頻度と類似度を考慮する方法である．関連語を算出するために単語の分散表現を使う．データ処理の流れを以下と図 1 に示す．

1. 各小説のあらすじを形態素解析ツール Mecab [2] で分かち書き文に変換する．
   - 新語対応のため形態素解析に IPA-Neologd 辞書 [3] を用いる．
   - Mecab での解析の際，分かち書き文に残す品詞を制限する．流行語は名詞が多いため，今回は名詞と固有名詞のみに制限する．
2. 分かち書き文書群から単語の出現頻度 (TF) を得る．TF のカウントには scikit-learn[4] を用いる．
3. 分かち書き文書群をコーパスとして FastText [5] に入力し，単語の分散表現を得る．
   - 分散表現（ベクトル）の次元数は 300 次元とした．

4　流行語抽出

本論文で検討した流行語の抽出方法を述べる．

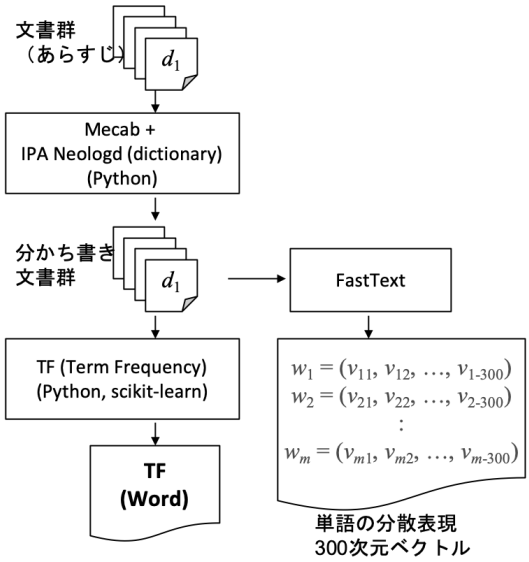

図 1: データ処理の流れ

4.1　方式 1:出現頻度のみ

方式 1 は単語の出現頻度のみで流行語を決めるもので，従来から用いられている素朴な方式である．質問文集合について，期間 p における単語 w の出現頻度 $tf(w,p)$ を求める．期間 p の出現頻度 $tf(w,p)$ が上位となる単語が，その期間の流行語である．

4.2　方式 2:類似単語の出現頻度を考慮

方式 2 として，類似単語の出現頻度も考慮する方法を提案する．1 つの物事を表す単語が 1 つしかない場合は少ない．省略語や類似する単語などで表現される場合が多い．例えば「オリンピック」と同様の単語に「Olympic」や「五輪」がある．意味的に近い単語に「オリパラ」がある.「オリンピック」の頻度に，類似単語の「Olympic・五輪・オリパラ」の頻度を加えることで,「オリンピック」の流行度をより良く計れるのではないかと考えた．

方式 2 では，期間 p における単語 w の出現頻度 $tf(w,p)$ に，w の類似語 t の値 $tf(t,p)$ を加える．ただし w と t の類似度 $sim(w,t)$ を乗じて加える．これを $new\_tf(w,p)$ とする．$new\_tf(w,p)$ の算出方法を式 (1) に示す．式 (1) の T は，あらすじに単語 w と共起出現する単語の集合である．

$$new\_tf(w,p) = tf(w,p) + \sum_{t \in T} sim(w,t) * tf(t,p)\ . \quad (1)$$

単語 w と t の類似度 $sim(w,t)$ は，fatstText が出力した単語の分散表現のコサイン類似度とする．fastText が算出する単語の分散表現（ベクトル）では，意味的に近い単語は近い値のベクトルとなることが多い．十分な文章量を持つコーパスを与えれば近いベクトルが出力されると期待できる．式 (2) にコサイン類似度の計算式を示す．

$$sim(w,t) = \frac{\sum_i v_{w,i} \cdot v_{t,i}}{\sqrt{\sum_i v_{w,i}^2}\sqrt{\sum_i v_{t,i}^2}}\ . \quad (2)$$

5　実験と考察

実験として，収集した小説メタデータのあらすじに対して方式 1 と方式 2 を適用した．なお流行語の推移粒度の期間 p は 1ヶ月ごとにした．方式 1 を適用した際のトレンドを表 2 に示す．2 つの期間（2010 年 10 月，2019 年 10 月）における出現頻度が上位の単語 10 個に限定して示し比較する．また方式 2 のトレンドを表 3 に示す．こちらも 2010 年 10 月，2019 年 10 月における new_tf 値が上位の単語 10 個を示す．表 2，3 にて（）内の数字は出現頻度，及び new_tf 値である．

表 2: 方式 1 の結果

	2010/10		2019/10	
Rank	単語	出現頻度	単語	出現頻度
1	の	1066.0	世界	5380.0
2	こと	882.0	の	4371.0
3	世界	736.0	こと	4040.0
4	人	476.0	異	2403.0
5	私	444.0	物語	1678.0
6	それ	440.0	彼	1616.0
7	彼	426.0	それ	1546.0
8	中	376.0	人	1509.0
9	物語	366.0	よう	1402.0
10	少女	362.0	主人公	1402.0

表 3: 方式 2 の結果

	2010/10		2019/10	
Rank	単語	出現頻度	単語	出現頻度
1	青年	2058.2	転移	14877.2
2	こと	1911.7	異	14272.7
3	事	1861.9	別世界	11702.4
4	中	1823.5	世界と日本	11687.4
5	少年	1670.0	世界文化	11066.4
6	同じ星	1638.8	新しい世界	10389.3
7	彼女	1631.1	世界	10291.8
8	辰原	1572.7	不思議な世界	8551.6
9	お付	1569.6	事	8497.1
10	転移	1566.5	現実世界	8449.5

表 2 を見ると，2 つの期間に出現する単語に大きな差がなく流行は掴めない．一方表 3 では，2010 年と 2019 年で出現する単語に大きな変化が見られる．2010 年は 10 位である「転移」が 2019 年では 1 位になっており，「異世界転生」と呼ばれるジャンルが大きく流行していることが分かる．しかし形態素解析処理の不具合から「世界と日本」などが名詞として認識されている．

6　流行語分析ツールの作成

本研究で作成した単語の出現頻度等のデータを利用した流行語分析ツールを作成した．本ツールは 2 つの単語と特定の年月を入力とし，それぞれの類似単語と類似度，指定した月での方式 2 における new_tf 値（[0,700] の範囲におさめて対数を取る）をグラフに出力する．単語同士があらすじの中でどれほど類似しているかを，類似語の観点から確認することが出来る．実際に「転生」と「魔法」の 2 単語と，2019 年 10 月を入力としたときのグラフを図 2 に示す．x 軸と y 軸はそれぞれの検索語に対する類似度である．グラフ上の円が左上と右下に分かれて分布しているため，「転生」と「魔法」はあまり似ていないことがわかる．対して「転生」と「転移」を入力とした場合のグラフを図 3 に示す．ここではグラフ上の円が右上に偏っているため，2 つの単語が類似していることがわかる．

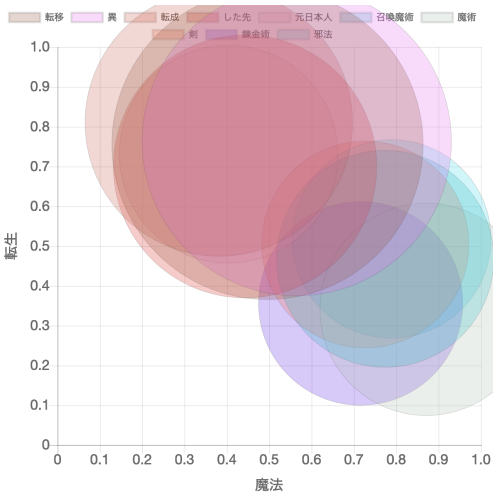

図 2: 「転生」,「魔法」を入力した場合のグラフ

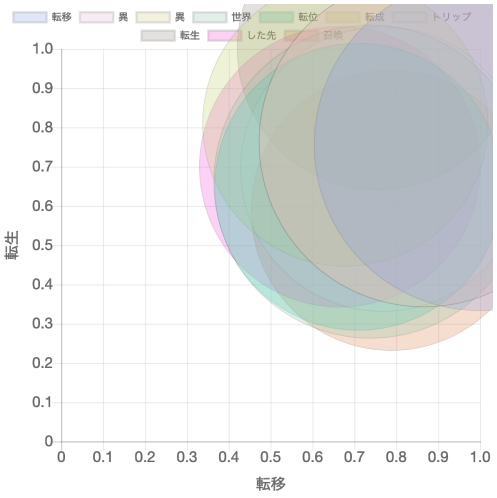

図 3: 「転生」,「転移」を入力した場合のグラフ

7　おわりに

本研究では単語の分散表現を用いた類似語抽出を用い，類似単語も考慮した流行語の抽出方式を提案した．実際になろう小説 API から取得した小説メタデータのあらすじを対象に提案方式を適用した．その結果，素朴な単語頻度による流行語抽出よりも，より意味を考慮した流行語抽出が出来た．流行語分析ツールも作成した．

今回は単語の分散表現を得るためのコーパスに小説のあらすじを用いた．Wikipedia 等の他のコーパスを用いた場合も比較したい．

参考文献

[1] `https://dev.syosetu.com/man/api/`. (Accessed on 12/30/2019).
[2] 松本裕治 Vol. 2004, No. 47, pp. 89–96 (2004).
[3] titleNeologism dictionary based on the language resources on the Web for Mecab (2015).
[4] E.Duchesnay Vol. 12, pp. 2825–2830 (2011).
[5] ArmandJoulinProceedings of the International Conference on Language Resources and Evaluation (LREC 2018) (2018).