オンライン小説の人気度推定に向けた検討

堺　雄之介[1,a)]　伊東 栄典[2]

概要：近年ネット上では，オンラインの小説投稿・閲覧サービスが人気である．人気が出たオンライン小説は，紙の小説本として書籍化されて販売されることも多い．大人気の小説は，漫画やアニメの原作になることもある．CGM 型のオンライン小説投稿・閲覧サイトでは，誰もが小説を投稿できるため，膨大な数の小説がサイト内に存在する．より早く，小説が人気になることを機械的に予測できれば，書籍化や漫画原作などのビジネスとして利益となる．読者には人気小説の検索や推薦が出来る．作者には，人気小説となるための指針を提供できる．本研究では，小説のメタデータ（タイトル・作者・あらすじ・キーワード）と小説本文の冒頭部分を用いて，その小説の人気度を推定する．本論文では，人気度の推定手法について報告する．

キーワード：小説, 機械学習, 推定

A Study for Estimating the Popularity of Online Novels

Yunosuke Sakai[1,a)]　Eisuke Ito[2]

Abstract: In recent years, online novel submission and reading services have become popular on the Internet. Online novels that have become popular are often sold as paperback books. Popular novels are sometimes used as the basis for manga and anime. Since anyone can post a novel on CGM-type online novel posting and browsing sites, a huge number of novels exist on the sites. If we can mechanically predict the popularity of a novel earlier, it can be profitable for businesses such as book publishing and comic book production. Readers can search for and recommend popular novels. For authors, the system can provide guidelines on how to make a novel popular. In this study, we estimate the popularity of a novel by using its metadata (title, author, synopsis, keywords) and the first part of the text. In this paper, we report the method of estimating the popularity and a small-scale experiment using data from "Shosetsuka ni Narou".

Keywords: Novel, Machine learning, Estimation

1. はじめに

近年，ネット上に動画・音楽・小説などのコンテンツを投稿するサービスが人気である．これらは利用者がコンテンツを投稿するサービスであり，CGM（Consumer Generated Media）と呼ばれる．CGM 型のサービスには YouTube や小説家になろうなどが存在する．これらのサービスには日々多数のコンテンツが投稿されており，利用者の数も膨大である．

CGM 型の小説投稿サービスでは誰もが小説を投稿できるため，数多くの小説が存在する．しかし，その中で人気を得る小説はごく一部である．そうした人気小説は印刷物として販売されたり，漫画やアニメの原作になることもある．より早く，小説が人気になることを機械的に予測できれば，書籍化や漫画原作などのビジネスとして利益となる．読者には人気小説の検索や推薦機能を提供できる．作者には人気小説となるための指針を提供できる．

そこで本研究では小説投稿サイト「小説家になろう」[1]の小説群を対象に，小説の人気度の推定を行う．人気度の推定には小説の本文の内，最初の数話を用いる．小説のブックマーク数を小説の人気度として利用する．

[1] 九州大学システム情報科学府
[2] 九州大学情報基盤研究開発センター
[a)] sakai.yunosuke.459@s.kyushu-u.ac.jp

2. 小説家になろう

「小説家になろう」は株式会社ヒナプロジェクトが提供する小説投稿サイトである．利用者は自由に小説を投稿・公開でき，公開された小説は誰でも読むことができる．2004 年に開設され，その当時は個人サイトとして運営されていた．アクセス増加により 2008 年からグループ運営へ変わり，2010 年に法人化した．Wikipedia によると 2019 年 4 月時点で，アクセス数が月間約 20 億，ユニークユーザーは約 1400 万人である．[2] 2021 年 1 月時点で利用登録者数は約 190 万人，掲載小説数は約 78 万件である．このサイトに投稿された小説は，サイト名から「なろう小説」と呼ばれることがある．

2.1 小説メタデータ収集

読者は小説を閲覧する際，小説の作者・題名・あらすじ・キーワード等を参考に読むかどうかを決める．こうした小説に関するデータをメタデータという．「小説家になろう」では小説のメタデータとして Ncode・タイトル・作者名・あらすじ・キーワード，またブックマーク数や人気尺度である総合評価点等が含まれる．本研究では小説のブックマーク数を小説の人気度として推定する．

小説メタデータの収集は「なろう小説 API」[3] を用いて行う．この API は株式会社ヒナプロジェクトが提供する Web API である．HTTP で GET リクエストを送ると JSON もしくは YAML 形式で，小説メタデータを取得できる．その際小説の ID にあたる Ncode を指定すれば，特定の小説のメタデータを受け取ることができる．

この API にリクエストを送り，全小説のメタデータを取得するクローラーを Python 言語で作成した．作成したメタデータクローラーの概要を図 1 に示す．

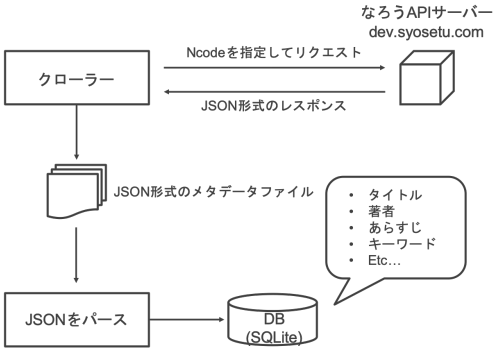

図 1 メタデータクローラー

2.2 小説本文収集

本研究では小説の本文冒頭部分を利用して人気度を推定するため，小説の本文を収集する必要がある．「小説家になろう」ではログインユーザーに対して本文ダウンロード機能を提供している．小説の ID である Ncode と話数を指定して，特定の小説の特定の話を取得できる．これを利用して小説本文を収集するクローラーを Python 言語で作成した．

クローラーで本文を収集する際，各小説の 2 話と 3 話を収集した．1 話には登場人物の紹介や相関図を掲載している小説が多く，これを本文とみなさないためである．またブックマーク数が 50 以上の小説に限定して本文を収集した．収集したデータの概要を表 1 に示す．

表 1 収集データ概要

項目	内容
期間	2004 年 4 月 20 日〜2020 年 12 月 1 日
小説件数	774,568
本文件数	44,191

2.3 収集データ分析

収集したメタデータ等について，可視化した結果を示す．以下のデータは本文を取得した 44,191 件の小説群を対象としている．

各小説のジャンルの割合を図 2 に示す．ファンタジージャンルが最も投稿されており，約 40%に及ぶ．人気度推定において，複数ジャンルの小説を対象にすると特徴を捉えられない可能性があるため，本研究ではファンタジージャンルの小説群を推定の対象とする．

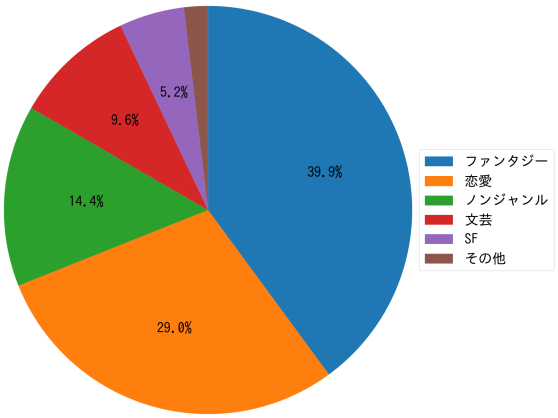

図 2 投稿小説のジャンル割合

「小説家になろう」におけるブックマーク数は，図 3 に示すような「ロングテール」と呼ばれるグラフになる．ここでブックマーク数の最大値は約 25 万であるが，ブックマーク数が 1 万を超える作品は上位 2 %程度である．また単語をベクトル化するためには文章を単語に分割する必要がある．形態素解析ツール MeCab[4] を用いて単語を分かち書きした時，対象とした小説群の単語数は平均 1 万単語である．また各小説の話数の平均は 72 話である．

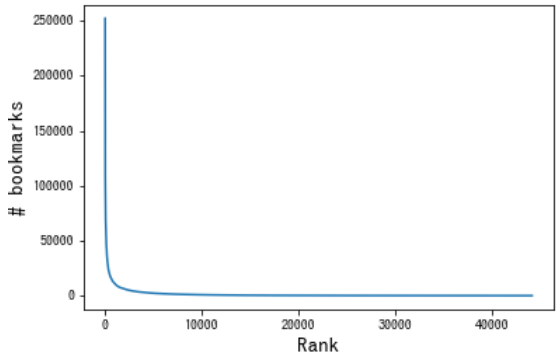

図 3 ブックマーク数と順位

3. 小説の人気度推定

本研究では小説のブックマーク数を小説の人気度として推定する．BERT を用いた機械学習による予測手法を説明する．

3.1 BERT を用いた予測器

機械学習で文書を扱う際, 文書をベクトル化する必要がある．単語の分散表現を獲得する手法として，Bag-of-Words や Word2Vec[5] などがあるが，今回は BERT[6] を用いる．

BERT は Google が 2018 年に発表した自然言語処理モデルである．BERT は Transformer[7] の Encoder を利用しており，Encoder では式 1 で表される Attention が用いられている．ここで Q,K,V はそれぞれ Query,Key,Value である．

$$Attention(Q,K,V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \qquad (1)$$

また BERT ではラベル無しコーパスを用いて事前学習を行う．事前学習においては Masked Language Model と Next Sentence Prediction の 2 つのタスクを学習する．Masked Language Model タスクではコーパスの一部を [MASK] トークンに置き換え，前後の文脈から元のトークンを予測する．Next Sentence Prediction タスクでは 2 つの文章が隣接しているものかを予測する．事前学習の後，ラベル有りデータを用いて転移学習を行うことで別のタスクに用いることができる．

BERT には単語を ID に変換した系列を入力として与える．入力する系列長は固定しなければならないため，最大系列長に満たない系列は [PAD] トークンで埋める必要がある．ここで [CLS] と [SEP] トークンが先頭と文章間に挿入される．BERT は各単語の分散表現を出力するが，[CLS] トークンに対する分散表現は，その文章の分散表現となる．

本研究では，BERT が出力した分散表現を全結合層に入力として与える．本研究におけるモデルの概要を図 4 に示す．

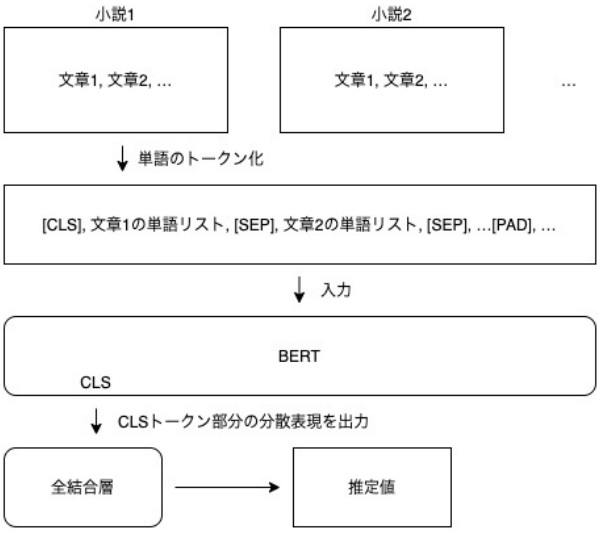

図 4 提案手法のモデル概要

3.2 予測器の評価

予測器の性能は平均二乗誤差（MSE），平均絶対誤差（MAE），交差検定を用いて評価する．

4. おわりに

本研究では，オンライン小説の本文を用いた人気度推定について述べた．小説投稿サービス「小説家になろう」の小説群を対象とし，ブックマーク数を人気度として推定する．小説データを収集するためにクローラを作成し，メタデータと本文を収集した．投稿される小説はファンタジージャンルのものが多く，ブックマーク数と順位はロングテールと呼ばれる形のグラフになる．

単語の分散表現を獲得するために BERT の事前学習済みモデルを用いる．その出力を用いて人気度を推定する機械学習のモデルを構築する．構築したモデルの評価には平均二乗誤差と平均絶対誤差を用いる．

今回は手法の提案のみで，実データにおける実験に至っていない．今後はデータを用いた実験を行う予定である．また Reformer[8] のような，非常に長い系列の処理が可能なモデルを用いた手法も検討したい．

参考文献

[1] ヒナプロジェクト社. 小説家になろう. http://www.syosetu.com/.
[2] Wikipedia - 小説家になろう. https://ja.wikipedia.org/wiki/%E5%B0%8F%E8%AA%AC%E5%AE%B6%E3%81%AB%E3%81%AA%E3%82%8D%E3%81%86.
[3] Narou-Developer. Narou api. http://dev.syosetu.com/man/api/.
[4] 拓工藤, 薫山本, 裕治松本. Conditional random fields を用いた日本語形態素解析. 情報処理学会研究報告. NL, 自然言語処理研究会報告, Vol. 161, pp. 89–96, may 2004.
[5] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *arXiv preprint*, Vol. arXiv:1607.04606, , 2016.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.
[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017.
[8] Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer, 2020.