

BERTで獲得する各場面の分散表現を用いたコサイン類似度に基づく小説の挿絵推薦

Shimokubo, Masato / 下窪, 聖人

(出版者 / Publisher)

法政大学大学院情報科学研究科

(雑誌名 / Journal or Publication Title)

法政大学大学院紀要. 情報科学研究科編

(巻 / Volume)

16

(開始ページ / Start Page)

1

(終了ページ / End Page)

6

(発行年 / Year)

2021-03-24

(URL)

<https://doi.org/10.15002/00023871>

BERT で獲得する各場面の分散表現を用いた コサイン類似度に基づく小説の挿絵推薦 Illustration Recommendation for Novels Using Cosine Similarity with the BERT based Distributed Representation of Each Scene

下窪 聖人

Masato Shimokubo

法政大学大学院情報科学研究科

E-mail: 19t0010@cis.k.hosei.ac.jp

Abstract

In recent years, E-book provides many opportunities that people read books during a commute or break time. Though an E-book reader provides some useful functions for reading the books. This system enables a new function of recommending illustrations during one is reading. This is achieved by calculating cosine similarity which uses the BERT based distributed representation of each scene. There are mainly 4 steps: (1) this system divides texts into some scenes by TopicTiling; (2) the important sentences are extracted by using BERT for each scene; (3) subjects, verbs, and objects are extracted using MeCab and CaboCha; and (4) the system recommends appropriate illustrations by calculating similarity between the sentences by using the analysis results and image titles. As expected, this system can recommend illustrations for a reader to form the image when he is difficult to understand E-book in only text content. These recommended illustrations are original from the illustrations of the web site “irasutoya”. As a result of the experiment, the accuracy of the illustration’s recommendation was about 72.0 percent. This result was obtained by using CNN to convert the image style of both the original and the recommended illustrations, and then evaluating them by SSIM.

1. まえがき

近年、電子書籍を通勤時間や休憩時間に活用する機会が増えている。このような場面において、読書をしながら用語を調べたり、知り合い同士で議論し合ったりすることがある。それらをサポートするために、Kindle などの電子書籍リーダーには読書支援ツール機能がある。例えば、小説の単語の意味を調べる機能、簡単に文字を拡大する機能、テキストを解析して小説の流れを可視化する機能などが挙げられる。これらの機能は自然言語処理分野で多く提案されている。しかし、小説を対象とした研究における共通の課題点として、テキストの文脈を正確に読み取ることが難しい点が挙げられている。この課題点を解決するために多く研究がなされているが、目立った実績はあげられてい

ない。本研究では、適切な挿絵を推薦することによって、ユーザのイメージ形成をサポートするシステムを提案する。本システムは場面分割をする TopicTiling [1]、重要文を抽出するための BERT [2]を用いる手法[3]、文章解析のための構文解析器 CaboCha と形態素解析器 MeCab を使用し、出力する挿絵は Web サイト「いらすとや」のイラストを使用する。本システムの出力結果は、シンプルな挿絵を推薦することを目標としている。本研究におけるシンプルな挿絵とは、複数の物体や人物などが描かれている挿絵の場合でも、その挿絵の中で最も重要である物体や人物に注目した絵と定義する。本システムの有効性を示すことができれば、挿絵の少ない小説に対して挿絵を与えることでユーザのイメージ形成をサポートできるようになる。挿絵が用意されていない場合、ユーザは自由度の高いイメージ形成を行う。そのため、著者の考えと離れたイメージを形成する可能性がある。本システムの推薦により、著者の考えを正確に追えるようになってと期待する。

2. 関連研究

2.1. 場面解析の研究

西岡ら[4]は、知的ナビ機能のついた電子書籍リーダーを提案している。この研究では、場面分割の代表的手法である TextTiling [5]と登場人物を対象にした固有表現抽出を使うことで場面を解析する。現在の電子書籍リーダーの読書支援ツールは、単一のキーワードをクエリとした検索は可能である。しかし、複数のキーワードをクエリとした検索はできないことが課題点として挙げられている。そのため、場面解析を行う提案システムを利用することで課題点を解決できることを示している。またこの課題点を解決することによって、場面ごとに複数の登場人物の出現を検索することが可能になることを示している。

2.2. 小説を対象とした推薦システムの研究

小坂ら[6]の研究はユーザの作品検索をサポートする推薦システムである。この推薦システムは作品の類似度を示す指標 S スコアと作品の質を示す指標 Q スコアを用いて推薦を行なっている。既存の推薦システムでは、読者が付与する評価値で推薦が行われている。そのため、読者が少ない小説は推薦されにくいという問題がある。この 2 つの

指標を予測するモデルを構築することでこの問題を解決している。さらに、ユーザが各指標の重みを調整できるように設計しているため、レコメンドの基準の操作も可能である。実験では Web サイト「小説家になろう」の小説を用いて、推薦システムの有効性を示している。

3. 挿絵推薦システム

本システムは、登場人物とその人物の行動を挿絵として示すことでユーザのイメージ形成をサポートする。本システムの構成を図 1 に示す。本システムの処理の流れは、テキストを図 1 の手順で解析し、その解析結果によって Web サイト「いらすとや」の画像を出力する。小説「浦島太郎」のテキストを入力として与えると図 2 のようになる。

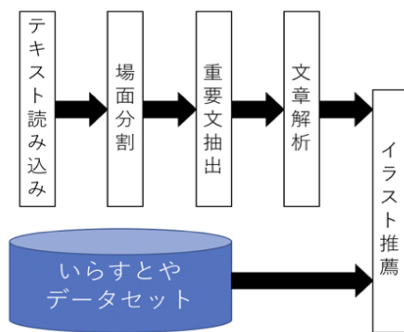


図 1 挿絵推薦システムの構成。

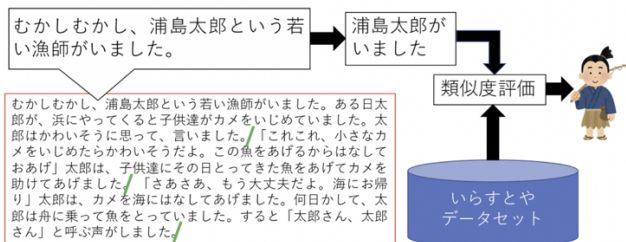


図 2 小説「浦島太郎」を入力として与えた結果。

3.1. 場面分割

小説や新聞は、ユーザの読書をサポートするために挿絵を挿入している。挿絵のないテキストのみの小説の挿絵を推薦するためには、テキストを解析して場面を分割し、各場面に対して挿絵を挿入するとユーザをサポートできると考える。テキストを場面ごとに分割するときは、作成したベクトルからそれぞれ区切り候補を求める必要がある。一般的に場面を分割するアルゴリズムとして、TextTiling [5]と TopicTiling [1]が使用されている。TextTiling [5]は、単語間の類似度と閾値を比較することで分割を行う。この手法の課題点は、1 文が短い文章が入力として与えられると、周辺の単語数が少ないことにより適切に場面を分割できないことが知られている。そこで、単語の出現情報を使用しない手法として、トピックモデルを使用する TopicTiling [1]が提案されている。TopicTiling [1]は、トピックモデルの Latent Dirichlet Allocation (LDA) を使用してトピックベクトルを作成し、場面境界の判定に使用する depth score を

求めている。LDA は、トピックごとに単語の出現確率の生成や、文章ごとにトピック確率の生成に使用されるモデルである。TopicTiling [1]の depth score は以下の式で求め、 p は位置、 c_p はある位置の coherence score、 $hl(p)$ は p より左側にある範囲で coherence score の最大値を返す関数、 $hr(p)$ は p より右側にある範囲で coherence score の最大値を返す関数である。本システムでは、1 文が短い場合にも対応できる TopicTiling [1]を使用する。また、coherence score はコサイン似度で求める。

$$depth\ score = \frac{hl(p) - c_p + hr(p) - c_p}{2} \quad (1)$$

3.2. 重要文抽出

挿絵は各場面の重要文をイラストとして表していると仮定すると、各場面の文章には内容と関係が薄い文が存在する可能性がある。重要文抽出によって関係が薄い文を取り除くことで、読者にとって必要な挿絵を推薦できると考える。重要文抽出は Yang Liu ら[3]が提案した手法を改良して使用する。Yang Liu ら[3]は、初めて文章要約タスクに BERT [2]を適用し、ニュース記事のデータセットで高精度の要約を実現した。Yang Liu ら[3]が提案した手法は英語のみに対応しているため、日本語の小説を処理できるように前処理を加える必要がある。本研究では、形態素解析器 MeCab を使って分かち書きをした後に入力として与える。また、日本語の学習済みモデルは日本語の Wikipedia の記事を学習したモデルを使用する。

3.3. 文章解析

登場人物は主語として必ず出現すると仮定する。このように仮定する理由は、小説は登場人物の行動や発言を中心に話を進める傾向があると考えたからである。本システムの文章解析は、各場面の重要文に対して構文解析と形態素解析を行う。これらの解析により、主語を key、目的語を含む述語を value とする辞書を出力する。例として「うらしまたろうはカメに乗った」を入力すると、辞書{うらしまたろう：カメに乗った}が出力される。処理の流れとして、まず構文解析で主語と述語を抽出する。ある文の主語が処理中に見つからない場合は、直前の主語をその文の主語とした。次に、形態素解析を行い、抽出された主語が適切であるかを判定する。主語が適切でないと判定された場合、その key を削除する。そして、形態素解析の処理で残った辞書を出力する。

3.4. イラスト推薦

文章解析後の辞書と Web サイト「いらすとや」のデータセットのラベルとの類似度を比較して、最も類似度が高いラベルを探索する。類似度を求める時には、BERT [2]で得られる分散表現とコサイン類似度を使用する。「いらすとや」のデータセットから、このラベルを持つイラストを出力する。この出力が本システムの出力結果となる。Web サイト「いらすとや」のデータセットは公開されていないため、スクレイピングによってデータセットを作成した。スクレイピングで収集した情報は、イラストとそのイラスト

トのタイトル（例：石に刺さった剣を抜くアーサーのイラスト）である。Web サイト「いらすとや」のイラストのタイトルには必ず末尾に「のイラスト」と記述されているため、類似度に影響を与える可能性がある。そこで、「のイラスト」の直前までをタイトル（例：石に刺さった剣を抜くアーサー）として保存する。また、Web サイト「いらすとや」には様々なカテゴリが存在している。本実験では、カテゴリ「物語」を対象として 219 枚のイラストを収集した。

4. 実験

本実験では、システムの評価を適切に行うため、イラストのある場面のみを対象とし、解析を行った。実験に使用するデータとして、青空文庫で公開されている小説 20 冊を使用する。青空文庫は、著作権が切れた文芸作品などを扱っていることが特徴として挙げられる。実験に使用した青空文庫のテキストの一部を図 3 に示す。類似度を算出する前に、CNN を使って画風変換を行うことでスタイルを統一した。元の挿絵との類似度が高ければ高いほど、元の挿絵の情報を多く抽出できているイラストであることを表す。青空文庫のテキストには、[# @@@] を基本形とする [# 3 字下げ] などのようにレイアウトやルビなどが含まれているため、削除した。

```

[# 挿絵 1 (fig51344_04.png、横487×縦575) 入る]
むかしむかし あるところに 4 ひきの こうさがが おりました。
なまえは それぞれ
[# 5 字下げ] フロブシー、
[# 4 字下げ] モブシー、
[# 3 字下げ] カトンデル、
[# 2 字下げ] ビーターです。
4 ひきは おかさんと いっしょに とってもおおきな モミのきの
したにある あなのなかに すんでいました。

```

図 3 テキストの一部。

4.1. 場面分割の評価

4.1.1. トピックモデルの評価

本実験に使用するトピックモデルは日本語の Wikipedia の記事で作成し、単語数は 83830910 語である。トピックモデルは一般的に Perplexity と Coherence で評価される。Perplexity は予測性能の評価として使われる。Perplexity の評価値は、正解の選択枝の数を表し、評価式を以下に示す。今回トピックモデルの作成に使用した単語数は 83830910 語であるため、評価値がこの単語数より小さければ小さいほど良いモデルとなる。

$$Perplexity = \frac{1}{P(\text{正解の単語} | \text{トピックモデル})} \quad (2)$$

Coherence はトピックの品質の評価として使われる。本実験では、Coherence を高精度で計算できる Shaheen ら[7] の評価方法を使用する。Shaheen ら[7]の方法は、まずテキストから単語ペアを作成する。作成後、単語ペアの確率を計算する。全ての単語ペアの確率の計算が終わった後に、ある単語集合と別の単語集合との自己相互情報量を計算

して、それを単語集合のベクトルとする。これらのベクトルでコサイン類似度を求め、コサイン類似度の平均を Coherence とする。この評価値が 1.00 に近ければ近いほど良いモデルとなる。

4.1.2. TopicTiling の評価

一致の誤差は前後 2 文とし、その範囲内であれば分割点が一貫していると仮定する。TopicTiling [1]での場面一致度は再現率、適合率、F 値によって評価する。TopicTiling [1]の既存手法である TextTiling [5]を提案した Hearst ら[5]の実験は、被験者が判断した場面境界と TextTiling [5]の一致度を再現率、適合率、F 値で評価を行っていた。そして、このときの F 値が 0.58 を超えていると、TextTiling [5]は十分に場面を分割できているという結果が得られていた。しかし、この方法は被験者の判断で結果が変わるため、客観的に評価することが難しいと考えた。そのため、本研究の評価は以下のように定義した式を使用した。挿絵を挿入している箇所とは、青空文庫のテキストにある [# 挿絵 n @.png 入る] (n : 文頭から数えた挿絵の番号, @ : 任意のファイル名) を指す。挿絵のルビの間に場面境界がある可能性も考えられるため、ルビ間の中央文と比較する実験も行なった。比較する手法として、場面分割の既存手法である TextTiling [5]を用意した。TopicTiling [1]は文間で場面を分割する手法であるが、TextTiling [5]は単語間で場面を分割する手法である。そのままの TextTiling [5]を使用する場合、TopicTiling [1]と TextTiling [5]を比較することができないため、TextTiling [5]が文間で場面を分割できるように拡張した。

$$\text{再現率} = \frac{\text{場面境界との一致数}}{\text{挿絵の数}} \quad (3)$$

$$\text{適合率} = \frac{\text{場面境界との一致数}}{\text{TopicTiling [1]の場面境界の数}} \quad (4)$$

$$F\text{値} = \frac{2 \times \text{再現率} \times \text{適合率}}{\text{再現率} + \text{適合率}} \quad (5)$$

4.2. 重要文抽出の評価

現在、テキストから重要文を抽出できているか評価する客観的手法は確立されていない。そのため、被験者に重要文を抽出させて F 値などで評価を行っていた。被験者を用意しない重要文抽出の一般的な評価方法として ROUGE アルゴリズム[8]が使われている。ROUGE アルゴリズム[8]は ROUGE-N, ROUGE-L, ROUGE-S など複数の手法を持ち、評価する重要文の集合と別の手法で用意した重要文の集合との類似度を求めて評価を行う。本研究の評価法は、N-gram (任意の N 文字が連続する文字列) の一致で評価する ROUGE-N と最長共通部分列の一致で評価する ROUGE-L を使用し、式を以下に示す。Count はある N-gram の出現頻度を求める関数である。N=1 と N=2 の結果が人間の評価に近いと言われている。LSC は一致する最長のシーケンス (共通部分列) を求める関数である。最長のシーケンスを求めることで最大限文脈を考慮することが可能になる。

$$ROUGE_N = \frac{\sum_{S \in S_{ref}} \sum_{Ngram \in S} Count_{match}(Ngram)}{\sum_{S \in S_{ref}} \sum_{Ngram \in S} Count(Ngram)} \quad (6)$$

$$\text{再現率} = \frac{LSC(Words_{reference}, Words_{system})}{Words_{reference}} \quad (7)$$

$$\text{適合率} = \frac{LSC(Words_{reference}, Words_{system})}{Words_{system}} \quad (8)$$

$$ROUGE_L = \frac{2 \times \text{再現率} \times \text{適合率}}{\text{再現率} + \text{適合率}} \quad (9)$$

実験データは Web サイト「小説家になろう」の小説 100 冊である。これらの小説は、1000 文以上 5000 文以下を満たしているものをスクレイピングで収集している。青空文庫の小説で実験を行わなかった理由は、青空文庫にはテキストデータのみが用意されており、小説の概要などの正解となる重要文が用意されていないからである。一方、Web サイト「小説家になろう」には、あらすじと本文が用意されている。しかし、このサイトの小説に挿絵は用意されていないため、重要文抽出の実験のみ採用した。本システムに採用した BERT [2]を用いる重要文抽出[3]は抽出型に分けられるため、比較する手法は抽出型の既存手法であるグラフ構造のみで重要度を評価する LexRank [9]を用意した。

4.3. イラスト推薦の評価

小説の挿絵と解析結果に基づいて推薦したイラストの類似度を求めて評価を行う。評価手法として、SSIM, MSE を使用した。このとき、挿絵と本システムが出力するイラストのスタイルを統一するために画風変換を行う。また、推薦した挿絵が文章解析後のテキストとどの程度類似しているのかを評価するためにコサイン類似度を使用する。本システムは、オブジェクトが 1 つのイラストを推薦しているが、青空文庫に用意されている挿絵には複数のオブジェクトが存在している。そのため、挿絵に複数のオブジェクトが含まれている場合は、1 つのオブジェクトを抽出した後に評価を行なった。また、Web サイト「いらすとや」に用意されているイラストは単色の背景画像であるが、青空文庫の挿絵には複数の色が使用されている背景が存在する。本実験では、青空文庫の挿絵の背景情報の影響を抑えるために、輪郭内を抽出して単色の背景画像に置き換えた。さらに類似度には正解の値が存在しないため、客観的に評価することが難しい。類似度が妥当であるか確認するために、多くのユーザが利用する Google の画像検索と本システムを比較する。Google 画像検索の検索結果は類似度が高い順に表示されることと、使用するユーザは十分に満足していると仮定する。

4.3.1. SSIM

SSIM は輝度、コントラスト、構造を軸にとることで、周囲の画素との相関を取り入れた指標である。SSIM の式を以下に示す。 μ_X と μ_Y は入力画像 X と Y の画素の平均、 σ_X^2 と

σ_Y^2 は入力画像 X と Y の分散、 σ_{XY} は入力画像 X と Y の共分散、 c_1 と c_2 は定数である。この評価指標は評価値が 1.00 に近ければ近いほど類似していることを表す。

$$SSIM(X, Y) = \frac{(2\mu_X\mu_Y + c_1)(2\sigma_{XY} + c_2)}{(\mu_X^2 + \mu_Y^2 + c_1)(\sigma_X^2 + \sigma_Y^2 + c_2)} \quad (10)$$

4.3.2. MSE

MSE は、オリジナル画像と推薦画像との差の二乗誤差を求める指標である。MSE の式を以下に示す。 X と Y は比較する画像の 2 次元配列、 M は画像の縦のサイズ、 N は画像の横のサイズである。この評価指標は評価値が小さければ小さいほど類似しているということを表す。

$$MSE(X, Y) = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (X(i, j) - Y(i, j))^2 \quad (11)$$

4.3.3. 画風変換

画風変換はコンテンツ画像にスタイル画像を取り入れる手法である。本研究では、色が異なる同じ名前の画像による実験結果への影響を最小限に抑えるために、スタイル画像の色を取り入れる。画風変換は、画像データセットで学習を行い、ニューラルネットワークの各層の出力値を利用する。そして、出力画像、コンテンツ画像、スタイル画像の損失を計算し加重和を減らすように更新を行う。

5. 実験結果

5.1. 場面分割の実験結果

5.1.1. トピックモデルの実験結果

TopicTiling [1]で使用するトピックモデルのトピック数を 2 から 1 ずつ増やして 4.1.1 項で述べた Perplexity と Coherence で評価する。まず Perplexity の結果について述べる。Perplexity の結果を図 4 に示す。図 4 より、トピック数が増えるにつれて Perplexity は増加する傾向にあることが読み取れる。

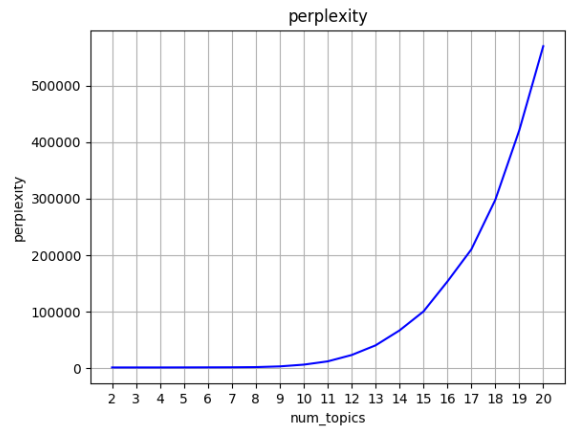


図 4 Perplexity の結果（トピック数 2～20）。

Coherence の結果を図 5 に示す。図 5 より、Coherence の評価値が大きい順に並べると、トピック数が 7, 3, 5, 12, ..., 16 であることが読み取れる。

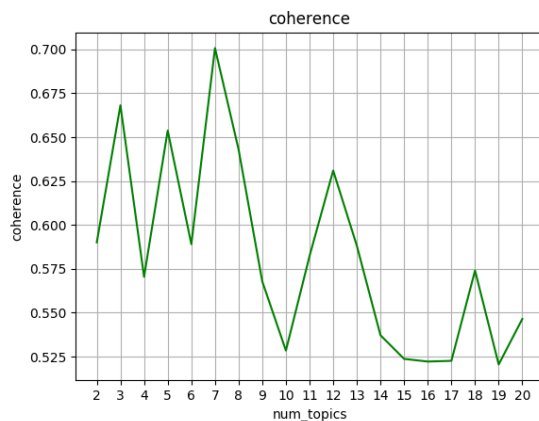


図 5 Coherence の結果（トピック数 2～20）。

トピックモデルは、Perplexity と Coherence の両方の評価値が良いトピック数を選ぶ必要がある。Perplexity の評価値のみが良い場合、トピックモデルに使用するトピックの品質が悪い可能性がある。逆に Coherence の評価値のみが良い場合、トピックモデルを使用した時の予測の性能が悪くなる可能性がある。図 5 の場合、トピック数が 7 の時に Coherence の最大値 0.70 を得ている。そのため、トピック数が 7 の時の Perplexity に注目する。トピックモデルの作成に使用したデータの単語数は 83830910 語であるのに対して、このときの Perplexity は 2145 である。すなわち、正解の選択肢の数を表す Perplexity は小さいため、Perplexity の評価値は良いと判断できる。したがって、トピック数が 7 のトピックモデルは、Perplexity と Coherence の両方の良い評価値が得られるモデルである。本システムの場面分割では、このトピックモデルを使用した TopicTiling [1]を使用する。

5.1.2. TopicTiling の実験結果

隣接する単語を含める範囲を決めるパラメータのウィンドウサイズを変更しながら TopicTiling [1]と TextTiling [5]を比較した結果を表 1 と表 2 に示す。表 1 と表 2 より、TopicTiling [1]は既存手法の TextTiling [5]を上回る結果が得られた。表 1 より、再現率、適合率、F 値の最大値を得るウィンドウサイズは 5 であることが読み取れる。そして表 2 より、ウィンドウサイズは 10 の時よりも 5 の時の方が最大値を得ることが読み取れる。表 1 と表 2 の評価値を比較すると、ウィンドウサイズが 5 の時と 10 の時の評価値はほぼ等しい。また図 3 より、文章の先頭にある挿絵のルビと話の始まりの部分が一致していることから表 1 の方が表 2 より適切であると考えられる。したがって、場面を分割した後に行われる重要文抽出、文章解析、イラスト推薦で扱うテキストは、ウィンドウサイズが 5 の TopicTiling [1]で場面を分割したものである。TextTiling [5]が TopicTiling [1]に劣った理由を述べる。拡張した TextTiling [5]は文の類似

度と閾値を比較して判定を行っていたが、各文の類似度がほぼ等しい場合適切に場面を分割することが難しくなる。各文の類似度がほぼ等しくなる理由として、文中に出現頻度が多い単語が周辺の単語に存在していれば、単語間の類似度に大きい変化がなくなり、各文の類似度に変化がないことが挙げられる。一方、TopicTiling [1]はトピックモデルを使用することで単語の出現情報を扱わないため、類似度が大きく変化し適切に場面を分割することが可能になった。

表 1 場面分割の実験結果（挿絵の挿入箇所）。

手法(ウィンドウ)	再現率	適合率	F 値
TopicTiling(5)	0.82	0.82	0.82
TopicTiling(10)	0.81	0.82	0.81
TopicTiling(15)	0.80	0.82	0.81
TextTiling(5)	0.46	0.77	0.50
TextTiling(10)	0.36	0.73	0.43
TextTiling(15)	0.40	0.72	0.45

表 2 場面分割の実験結果（ルビ間の中央文）。

手法(ウィンドウ)	再現率	適合率	F 値
TopicTiling(5)	0.88	0.88	0.88
TopicTiling(10)	0.87	0.88	0.87
TopicTiling(15)	0.86	0.87	0.86
TextTiling(5)	0.46	0.86	0.52
TextTiling(10)	0.39	0.85	0.48
TextTiling(15)	0.48	0.85	0.54

5.2. 重要文抽出の実験結果

実験結果を表 3 に示す。表 3 より、BERT [2]を用いる手法[3]が既存手法の LexRank [9]を全て上回っている。すなわち、本システムの重要文抽出に採用した手法は、既存手法より有効である。LexRank [9]は、コサイン類似度でグラフを構築するため、各文の共通情報を持つ簡潔な文を抽出する可能性がある。この課題点が評価値を下げたと考えられる。ニュース記事で実験を行なった尾崎ら[10]の結果に近い評価値が得られていることと、BERT [2]を用いる手法[3]は既存手法を上回ることが確認できた。本実験で使用したデータは小説のテキストであるため、全体の流れを考慮して重要文を判断する必要がある。しかし、本システムに使用している手法は、全体の流れを考慮して重要文を抽出するのではなく、学習済みモデルから重要文を予測して抽出する。この課題点を解決する方法として、登場人物の関係や行動から場面の变化を高精度で判定できることが必要であると考えられる。

表 3 「小説家になろう」100 冊の平均。

	ROUGE-1	ROUGE-2	ROUGE-L
BERT	0.33	0.06	0.18
LexRank	0.28	0.03	0.16

5.3. イラスト推薦の実験結果

実験結果を表 4 に示す。比較する分散表現として Doc2Vec を用意した。SSIM と MSE の評価値が減少した原因として 3 つ挙げる。1 つ目は、複雑なオブジェクトを含

む挿絵から1つのオブジェクトを抽出する方法が、複数のオブジェクトを完全に削除できずに残った状態で出力されている可能性がある。2つ目は、イラスト内のオブジェクトの向きが考慮されていないことである。解決する方法として、オブジェクトの向きを統一する方法が考えられる。この方法は、挿絵のオブジェクトの方向を自動的に判定する必要がある。しかし、イラスト内のオブジェクトの向きを推定する有効な方法は提案されていないため、今後の課題点となる。3つ目は、重要文抽出の影響を受けている可能性がある。5.2節の結果では、重要文抽出が有効に機能していないことが示されている。有効に機能しない場合、挿絵に必要な情報を抽出することは困難になる。そのため、不必要な情報で挿絵の推薦が行われた可能性がある。この問題点を解決するには、小説のテキストを解析することで、場面の変化を読み取り、重要文を適切に抽出する手法が必要になる。表4のMSEより、Doc2Vecの方がBERT[2]より誤差が少ないことが読み取れる。すなわち、Doc2VecはBERT[2]より挿絵の色に近いイラストを推薦していると考えられる。コサイン類似度の実験結果について述べる。表4のコサイン類似度より、BERT[2]のコサイン類似度は0.79であることに對して、Doc2Vecのコサイン類似度は0.69であり、類似度は減少した。減少した理由として、Doc2Vecの単語の内部状態が文脈によって変化しないことが挙げられる。次に、重要文抽出の影響を確認するために、重要文抽出を行わなかった結果と比較すると、評価値にはほぼ影響を与えていない。そのため、重要文抽出はイラスト推薦に対して有効に機能していないと考えられる。最後に、Googleの類似画像検索と比較すると本システムは、SSIMでは上回っているが、MSEは劣る結果となった。したがって、本システムは正解の挿絵に対して、構造に類似している画像を推薦できているが、色に類似している画像は推薦できていない。また、コサイン類似度の結果より本システムは解析後の文に十分に近いイラストを推薦していると考えられる。

表4 青空文庫20冊の評価値の平均。

	重要文抽出	分散表現	SSIM	MSE	cosine
提案手法	あり	BERT	0.72	254.18	0.79
	あり	Doc2Vec	0.72	246.61	0.69
	なし	BERT	0.72	254.28	0.78
	なし	Doc2Vec	0.72	247.66	0.69
Google 類似画像検索		BERT	0.62	209.49	0.66
		Doc2Vec			0.46

青空文庫のテキストが本システムに入力されてから挿絵が推薦されるまでの実行時間の結果を表5に示す。このときの実行時の環境は macOS High Sierra (メモリ 16GB, プロセッサ 2.6GHz) である。表5より、実行時間の速い順にソートすると、Doc2Vec (重要文抽出なし)、Doc2Vec (重要文抽出あり)、BERT[2] (重要文抽出なし)、BERT[2] (重要文抽出あり) という順になる。このことから、BERT[2]よりも Doc2Vecの方が速く推薦できると考えられる。速度を優先する場合、Doc2Vecは多義語を解釈できないため、推薦の精度は落ちてしまう可能性がある。

表5 本システムの平均実行時間。

	重要文抽出あり		重要文抽出なし	
分散表現	BERT	Doc2Vec	BERT	Doc2Vec
平均[sec]	125.55	55.06	91.88	16.15

6. むすび

本研究では、BERT[2]で得られる分散表現に対してコサイン類似度を計算することで適切な挿絵を推薦するシステムを提案した。青空文庫の小説20冊で実験を行なった結果、本システムは十分な精度で挿絵を推薦していることと、BERT[2]の分散表現を使用して計算することが有効に機能していることが示された。さらに精度を向上させるためには、重要文抽出を有効に機能させる必要がある。本実験では、BERT[2]の分散表現やDoc2Vecの分散表現を用いて重要文を抽出していた。しかし、これらの分散表現は内容に最も関係している文を見つけることが困難であるという結果が得られた。人物の相関図などのテキスト情報以外の情報も機械に与えることで、この問題を解決できる可能性がある。そのため、テキスト以外の情報が重要文抽出にどの程度影響を与えるのかについて調査を進める。

文 献

- [1] Martin Riedl, Chris Biemann, “TopicTiling: A Text Segmentation Algorithm based on LDA”, *Proc. ACL 2012 Student Research Workshop*, pp.37-42, Jul.2012.
- [2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”, *arXiv preprint arXiv:1810.04805*, Oct.2018.
- [3] Yang Liu, Mirella Lapata, “Text Summarization with Pretrained Encoders”, *Proc. EMNLP-IJCNLP*, pp.3730-3740, Nov.2019.
- [4] 西岡 渉, 沼尾 雅之, “小説を対象とした知的ナビ機能の付いた電子書籍リーダの提案”, *DEIM Forum 2016 E8-5*, Mar.2016.
- [5] M. Hearst, “Multi-Paragraph Segmentation of Expository Text”, *Proc. the 32nd Annual Meeting of Association for Computational Linguistics*, pp.9-16, Jun.1994.
- [6] 小坂 直輝, 小林 哲則, 林 良彦, “隠れた良作を推薦可能なWeb小説レコメンドシステムの提案”, 人工知能学会インタラクティブ情報アクセスと可視化マイニング研究会 (第23回) 研究発表予稿集, pp.1-8, Nov.2019.
- [7] Shaheen Syed, Marco Spruit, “Full-Text or Abstract? Examining Topic Coherence Scores Using Latent Dirichlet Allocation”, *Proc. the 4th IEEE International Conference on Data Science and Advanced Analytics*, pp.165-174, Oct.2017.
- [8] Chin-Yew Lin, “ROUGE: A Package for Automatic Evaluation of Summaries”, *Workshop on Text Summarization Branches Out, Post Conference Workshop of ACL 2004*, pp. 74-81, Jul.2004.
- [9] G. Erkan, D. Radev, “LexRank: Graph-based Lexical Centrality as Saliency in Text Summarization”, *Journal of Artificial Intelligence Research* 22, pp. 457-479, Dec.2004.
- [10] 尾崎 花奈, 小林 一郎, “事前学習モデルと潜在トピックを用いた文書要約への取り組み”, 言語処理学会第26回年次大会発表論文集, pp.828-831, Mar.2020.