DEIM Forum 2021 A25-2

あらすじの抽象的グラフ化による似た物語展開を含む映画の検索

氏家　翔馬† 　栗原　光祐†† 　荘司　慶行† 　MartinJ. Dürst†

† 青山学院大学 理工学部　〒 252 – 5258 神奈川県 相模原市 中央区 淵野辺
†† 青山学院大学理工学研究科理工学専攻知能情報コース　〒 252–5258 神奈川県相模原市中央区淵野辺 5–10–1
E-mail: †{ujiie,kurihara}@sw.it.aoyama.ac.jp, ††{shoji,duerst}@it.aoyama.ac.jp

あらまし　本稿では，グラフエンベッディング技術を用いることで，固有名詞などによらず，物語の展開の類似した映画を発見するアルゴリズムを提案する．具体的には，ある任意の映画を入力した際に，登場人物名や出演者の一致した映画ではなく,「大人と子供がタッグを組んで逃げる」など，類似した展開を含む映画を検索可能にする．そのために，1）単語を分散表現で表しクラスタリングし，表現を統一することで固有名詞の影響を排除できる，2）あらすじ文をグラフ化し類似度計算することで，表現レベルのぶれやあらすじ文の記述粒度の影響を受けづらくできる，という 2 つの仮説に基づいてアルゴリズムを作成した．実験のために，Yahoo!映画にある映画の一覧と，Wikipedia のそれらに対する記事のあらすじから作成したデータセットを用いて，Word2Vec と kNN による抽象化, 係り受け解析と graph2vec によるグラフ化による検索システムを実装した．実際に映画の類似度を判定する被験者実験を行うことで，それぞれの仮説がどのように働くかを確認した．

キーワード　情報検索，あらすじ検索，グラフエンベッディング

1 はじめに

定額配信サイトの普及や，社会的状況にもとづく巣篭り需要の増大に伴い，インターネット上で映画を視聴する機会が増えてきている．このようなインターネット上の視聴では，従来の映画館での視聴の際には事前に視聴したい映画を十分に吟味するのに対して，より気軽に新しい映画を予備知識のない状態で視聴する場合が多い．

このように，特に興味のない映画を何となく視聴し，その映画が気に入った場合のことを考える．ある気に入った映画があった場合に，似たような映画をもっと見たいと感じたとする．従来の類似度に基づく映画検索では，メタデータや登場する語に基づいて類似度が計算されることが一般的である．そのため，監督や俳優が重複していたり，登場人物名などの固有名詞が近い映画が多く発見される．

一方で，このような場合には，同じような物語の展開を持つ映画が発見したいという場合も多い．監督，俳優，ジャンル，シリーズ作品の情報と同様に，その映画がどのような物語展開をもつかも，次に視聴する映画を決定するうえで非常に重要な要素である．しかし，現状の映画批評サイトでは同じシリーズの作品や，監督や出演している俳優が他に参加している映画を検索することは可能だが，物語の展開が類似した映画を検索する手段が用意されていない．

具体的には，映画「ターミネーター 2」を視聴し，類似した映画を検索する際に,「ターミネーター 3」などのシリーズ作，同じ監督の「タイタニック」や，同じ出演者の「コマンドー」を探すことは可能である．しかし，例えば「子供と大人がタッグを組んで敵から逃げる」という似たあらすじを持つ映画を発見することはできない．映画「ターミネーター 2」と類似した作品として,「レオン」や「グロリア」などの「子供と大人がタッグを組んで敵から逃げる」という，似たあらすじを持つ映画を見つけるためには，あらすじを逐一読んで，人手で比較しなければならない．

映画は視聴者の習熟度の差異が大きい分野であり，映画情報の検索アルゴリズムも，幅広い利用者にこたえる必要がある．日常的に映画を視聴する利用者もいれば，年に 1 回映画を見るか見ないかというような，映画に詳しくない利用者もいる．映画に詳しい利用者であれば，監督名やジャンル名から映画の傾向を予想して次に見る映画を決めたり，公開中の映画の情報を逐一チェックしたり，映画好きのコミュニティ内で情報を共有することが可能である．しかし，近年の動画配信などで気軽に映画を見るような視聴者は，監督や俳優，作中の固有名詞に関する先行知識がない場合も多い．そのため，各映画情報サイトの関連映画を検索し情報を確認しても，次に見たい映画を満足に検索することができない．

映画の先行知識がない状態でもあらすじの類似した映画を検索可能になることで，日常的に映画を視聴しないユーザは，より多くの新しい映画を発見し，映画に詳しくなっていくことができる．また，既に映画に詳しいユーザにとっても，これまで視聴していなかった作品を発見でき，興味を広げられると考えられる．

このような，メタデータや固有名詞に依らない，あらすじの類似度に基づく類似映画の検索を可能にするために，

- あらすじに含まれる語の抽象化，
- あらすじのグラフ化とベクトル化

というふたつの処理を行う．

あらすじに含まれる語の抽象化は,「語の表現を統一することで固有名詞の影響を排除できる」という考えに基づく．具体的には「フィンランド」や「アイルランド」という単語を,「北欧

の国」といったような意味を持つラベルに置き換える．そのために，単語を文字列レベルで統一し，その後に分散表現化したうえでクラスタリングする．実際の語の代わりに，どのクラスタに属すか（すなわち，クラスタ番号）を類似度比較に用いることにより，固有名詞に依らない文同士の検索が可能になると考えられる．

あらすじのグラフ化とベクトル化では，あらすじ文をグラフ化し類似度計算することで，表現レベルのぶれやあらすじ文の記述粒度の影響を受けづらくできるという考えにもとづく．

実際のアルゴリズムとして，映画のあらすじ文を係り受け解析し，語をノードとする係り受けグラフとして表現する．次に，データセットに含まれるすべてのノードの語について，文字列として抽象化する．次に，各語を分散表現化し，クラスタリングによって実際の語からクラスタ番号に置き換える．こうして，クラスタ番号をノードとするあらすじグラフについて，グラフエンベッディング技術を用いてベクトル化する．こうして作成されたグラフの分散表現ベクトルについて類似度を計算することで，映画のあらすじが意味的に近い順に映画をランキングできる．

実際に，Yahoo!映画に含まれる映画情報と Wikipedia におけるあらすじの記述からデータセットを作成し，被験者実験を通して手法の有用性を評価し，それぞれの仮説がどのように作用するかを分析した．

本論文の構成を記す．本論文は, 本章を含め全 6 章からなる．本章では, 本研究を行うに至った背景と研究の目的に述べた．第 2 章では, 本研究に関連した研究について紹介する．第 3 章では, 提案手法について述べる．第 4 章では, 使用したデータとあらすじ文を用いた映画検索, 評価について述べる．第 5 章では, 実験結果について考察する．第 6 章では, 本研究のまとめと今後の展望を述べる．

2 関 連 研 究

本研究は映画のあらすじを検索するために，登場する単語を抽象化し，またあらすじ文をグラフ化して用いている．そこで，語の分散表現化と抽象化について 2.1 節で，グラフエンベッディング技術について 2.2 節で，あらすじ検索について 2.3 節でそれぞれ関連研究を紹介し，論じる．

2.1 語の分散表現と抽象化

語の抽象化とは，表記は全く異なるが同じ役割を持つ単語を，抽象化することで統一する手法である．具体的には,「ターミネーター 2」の登場人物であるジョンと「レオン」の登場人物であるマチルダは,「大人とタッグを組んで敵から逃げる子供」という共通した役割を持っている．両者を役割にそって「子供」として統一することが，語の抽象化である．

最も単純で古典的な，語の表現を統一する方法として，レンマ化が挙げられる．これは，文字列に決められた前処理を行って語を統一する手法である．レンマ化の具体例として，Plisson ら [1] はスロベキア語において，リップルダウンルールを用いて接尾語の操作を行い，語を統一する研究を行っている．しかし，これらの手法では語を文字列的に共通化しているだけなので，語の統一には不十分である．近年では語の意味を考慮する語彙表現手法として，語の分散表現が用いられている．古典的な手法では，TF-IDF によって算出した語の出現頻度を用いて，LDA（Latent Dirichlet Allocation）や LSI（Latent Semantic Indexing）で次元圧縮を行って類似した語同士をまとめていた．近年では，Word2Vec [2] や Doc2Vec [3] などの，前後の単語から該当単語を予測する単層ニューラルネットワークを用いた語の分散表現化が一般化してきている．より一般的な手法として，BERT [4] は，大規模なニューラルネットワークで学習したモデルをデータにあわせてファインチューニングして用いる．本研究では，コーパス数の少ない映画のあらすじにあわせて逐一再学習を行いたいため，計算機資源のことを考慮して Word2Vec を使用した．

Hijikata ら [5] は，あらすじ文に含まれやすい単語を用いてスコアリングし，入力した文章があらすじ文であるかを判別する研究を行っている．この研究では，未知語となる登場人物名を抽象化することで分類精度を向上しており，単語の抽象化の有効性を確認している．この研究では，人名の抽象化に既存の辞書を用いて，人物名を一般化している．本研究では，対象が新作の公開され続ける映画であるため，自動で計算可能な単語の分散表現を用いる．Anandan ら [6] は医学分野における症例の共有の際に，必要な固有名詞を削除しなければいけない問題に対して，固有名詞を一般化することで文意に沿った情報の削除を行っている．このような人名の置き換えは，例えば Sweeney ら [7] の研究のように古くからお行われている．

2.2 グラフエンベッディング

グラフエンベッディングとはグラフを分散表現化する手法である．具体的には，あるグラフ中のノードや，グラフ中の部分グラフを分散表現化することで，機械学習や計算に使用できるようにする．本研究では，あらすじ文をグラフ化したのちに分散表現化し，類似度計算することで，似た物語展開を持つ映画を検索可能にしている．

グラフエンベッディングの代表的な例として，node2vec [8] が挙げらる．文書における単語のエンベッディング手法である Word2Vec で用いられる Skip-gram モデルを利用して，グラフ内の各ノードの分散表現を算出している．ノードを分表現化する際には，Random Walk が用いられることが多い．これは，特定のノードから設定した深さのノードをランダムに探索する手法である．node2vec は，Random Walk に特定のパラメータを設定することで幅優先探索と深さ優先探索のどちらを行うかを設定可能としている．

本研究で利用した graph2vec [9] は文書における文のエンベッディング手法である Doc2Vec の PV-DBOW の考え方を拡張し，グラフの分散表現を算出するアルゴリズムである．node2vec がノードを分散表現化するのに対し,graph2vec では部分グラフを分散表現化することができる．グラフ ID を入力として，グラフに登場する部分グラフを予測し，その成否によって内部処

理を変更することで分散表現を算出するニューラルネットワークである．Weisfeiler-Lehman（WL） Relabeling 戦略を流用しており，グラフの形状に特に注目して分散表現化を行っているのが特徴である．本研究では，グラフの分散表現を算出する際に使用した．

グラフ化の際のノードに対するラベリングについて，Bhagat ら [10] はランダムウォークや反復法を拡張しているが，本研究ではノードとなる単語の分散表現をラベリングし，グラフに付与している．

グラフエンベッディングを物語グラフに用いる例として，Lee [11] らは登場人物の関係や状況を抽出したグラフの分散表現を比較しているが，本研究ではあらすじ文に沿ってグラフを作成し，登場人物の抽象化を行い分散表現化して比較している．

Li ら [12] はストーリーをグラフで表し，ノードの共起条件などを設定して自動であらすじを作成している．

2.3 あらすじ検索

あらすじ検索とは，あらすじ文の一部などをクエリとして，目的に一致したあらすじを持つ作品を検索することである．本研究では映画のあらすじを対象として，クエリとして入力されたあらすじ文に類似したあらすじを持つ映画を検索している．このような研究は，これまでにも多く行われてきた．Kyozuka ら [13] はあらすじ文の一部をクエリとしたあらすじ検索手法を提案している．この研究では語の削除を役割によって段階的に行っているが，本研究では品詞によって一括で削除している．Xiong ら [14] は，ビデオの目的シーンを検索するために，シーンを人物，場所，イベントなどの情報を使用してストーリー形式にしている．Park ら [15] は，登場人物の関係を使用して映画のストーリーを検索する研究を行っている．Billsus ら [16] は，ニュースをあらすじで分類し，ユーザが興味を示す内容のみ提示する研究を行っている．Solomon ら [17] は，ヒンディー語とテルグ語の物語を，導入，メイン，クライマックスのみっつの部分に分けて分析し，寓話，民話，伝説に分類する研究を行っている．本研究では，これらの分類を機械学習を用いることで自動化し，任意の物語と似た類型を持つ別の物語を検索可能にすることを目的としている．

3 手　　法

映画をあらすじ文を用いて検索する手法について述べる．物語の展開が似た映画を検索するために，自然言語で書かれた文をグラフ化する．そして, グラフを構成するノードであるそれぞれの単語について, 文字列レベルでの表現の統一と, 意味レベルでの統一を行う．こうして作成されたノードが抽象化されたラベルに置き換わったグラフについて, グラフエンベッディング技術でベクトル化し, 類似度比較を行う．

3.1 あらすじ文のグラフ化

あらすじ文の似た映画を探す際に，あらすじの記述の粒度や文章の特徴を排除するために，あらすじ文のグラフ化を行う．

あらすじ文において, 主語と目的語の関係や, 文章の時系列は

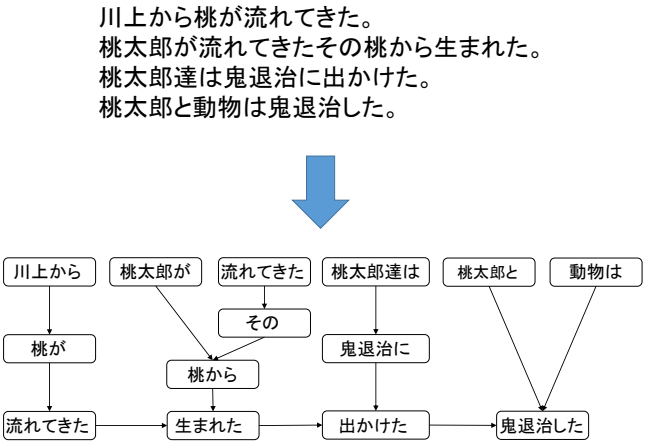


図 1　係り受け解析による実際のあらすじ文をグラフ化の例. 各文を木構造で表し, ルート同士を辺で結んだ.

特に重要である．そこで作成するグラフはエッジの向きによって時系列情報を保持できる有向グラフを用いた．グラフ化は 1 本の映画ごとに, あらすじ文単位で行った．あらすじ文を係り受け解析し，節をノード・リンクをエッジとしてグラフ化した．そして，各グラフの根ノードを時系列順にエッジで繋いだ．この操作によって，あらすじ文の時系列に沿った 1 本の映画のグラフを作成した．実際のグラフ化の例を図 1 に示す．図 1 の上部の 4 文で構成されるあらすじが，あらすじ文の構成に沿ってグラフ化されている．

注意事項として，本研究では，あらすじをグラフ化する際に，係り受け解析結果をグラフとして用いた．既存研究として, プロットグラフと呼ばれる物語を正しくグラフとして表す手法が提案されている．しかし, これらのグラフは自動で作成できず, 作成にはクラウドソーシングなどの人手での作業を要する．現時点では物語展開そのものを正しく自動でグラフ化する手法が確立されていないため，本研究では文章構造をグラフ化する係り受け解析を用いた．

3.1.1 文字列レベルでの単語表現の抽象化

本研究で用いているあらすじ文には，同じ対象を指していても表現の異なる語が含まれる．そのため，検索精度の向上のために文字列レベルでの単語表現の統一を行う．具体的には，

- サ変活用以外の動詞・非自立以外の名詞・形容詞のみを抽出，
- ノード内に名詞が登場した場合，以降の動詞を削除，
- ノード内に動詞が複数登場した場合，2 個目以降の動詞を削除，
- ノード内の単語の原型化

という処理をそれぞれ前処理として行った．これらの前処理の際に，節内の単語がすべて削除された場合，その節を無いものとして扱う．

また，固有名詞の中に「・」が含まれる場合は「・」で分割し，あらすじ文中で登場する同じ対象を指した語に置き換える．例えば，サラ・コナーという固有名詞は 2 度めの登場以降はサ

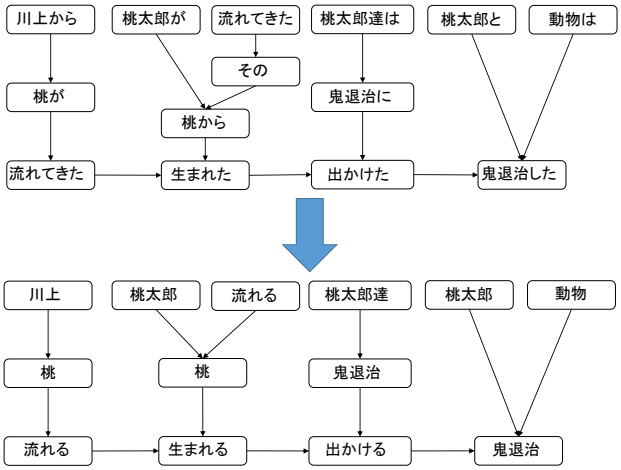


図 2　実際にグラフ上で表現の統一を行った例．各ノードの表現が統一され, 不要な情報の削除とエッジの再構成が行われている．

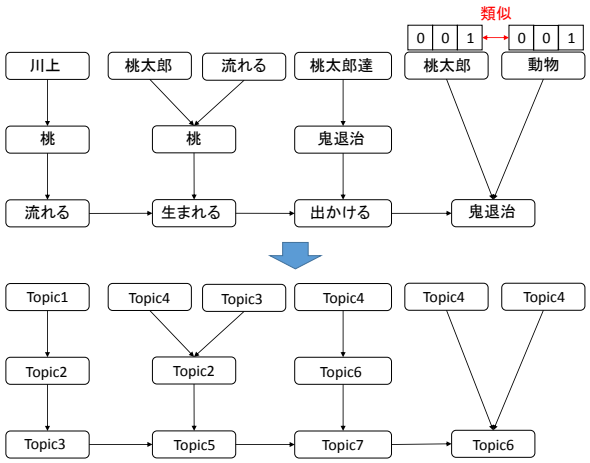


図 3　Word2Vec で類似した意味の語同士を同一のラベルに置き換えた例．

ラと表記されるので，サラ・コナーをサラと置き換えることによって表現を統一した．図 2 に実際にグラフのノードに対して処理を行った例を示す．前処理によって余分な情報が除去されているのが分かる．

3.1.2　意味的レベルでの単語表現の抽象化

ここまでで，文字列レベルで表現を統一できた．次に，表記の全く異なる名詞が近い意味を持つ際に統一する．具体的には，単語の分散表現を用いて類似した語をまとめて 1 つのラベルとして表し，グラフの各ノードをラベルを用いて表現することで，意味的レベルで表現を統一した．

最初に，各節の単語を分散表現化するために学習用のコーパスを作成した．

Word2Vec [2] を用いるため，使用する文章を分かち書きしたものをコーパスとして用いた．学習用のコーパスには，係り受け解析と不要な単語の削除を行った映画のあらすじ文に含まれる約 14 万種の単語と，映画以外も含む約 20 万件の Wikipedia の記事から抽出した約 30 万種の単語が含まれる．Wikipedia 記事内の単語については，1 度しか登場しない単語は正確な学習の妨げとなるため，コーパスから取り除いた．また，Word2Vec は前後の単語を用いて学習を行うことを考慮し，コーパスは記事の章が変わるごとに改行した．この処理によって，異なる章の単語を学習に使用し，学習の精度が低くなることを防ぐ．作成したコーパスを用いて学習し，各単語を分散表現化した．映画のあらすじに含まれる 14 万種の単語の分散表現を用いてクラスタリングし，各単語をラベルで表現する．図 3 に実際にラベルを用いてグラフを構成する例を示す．分散表現が近似している単語を同一のラベルに変換している．

実際の計算の際には，同一のラベル表記のノードをまとめて計算するため，同一のラベル表記のノードは統一してグラフ化した．ラベル表記を統一したグラフの例を図 4 に示す．

3.1.3　グラフの分散表現を用いた映画のあらすじの類似度比較

本研究では, 文をグラフ化することで記述の粒度や文体の影響を減らせると考えたため, あらすじ文をグラフとして扱った．実際にグラフ同士の類似度を計算するために, グラフをベクトルに変換する．

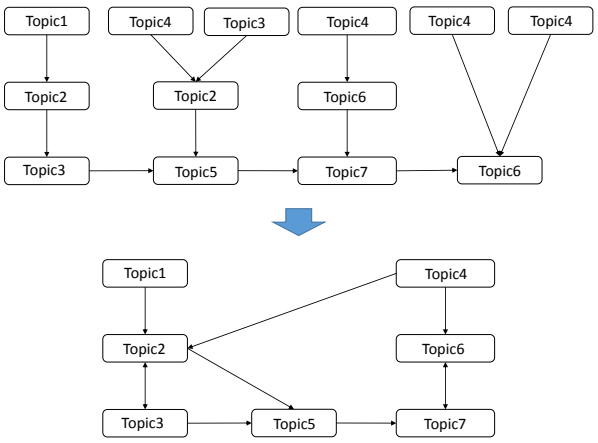


図 4　同一のラベルを持つノードを単一のノードとして集約した例．

各グラフを graph2vec [9] を用いて分散表現化した．graph2vec とは，Doc2vec の PV-DBOW の機能を拡張し，グラフ ID を入力してグラフ中の部分グラフが登場する確率を予測する単層ニューラルネットワークである．グラフ ID にかける重み行列の各行が各グラフの分散表現である．グラフの分散表現の類似度を用いて，映画のあらすじ文が類似した映画を分類した．

図 5 に実際にグラフの分散表現を用いて映画の類似度を算出した例を示す．あらすじ文のグラフの分散表現のコサイン類似度を算出し比較した．

4　評価実験

本章では，あらすじ文のグラフ化と単語の抽象化が，映画のあらすじ文の検索精度向上に有効であるという仮説を検証するため，ベースラインと比較手法を含む 5 つの手法で実験を行う．

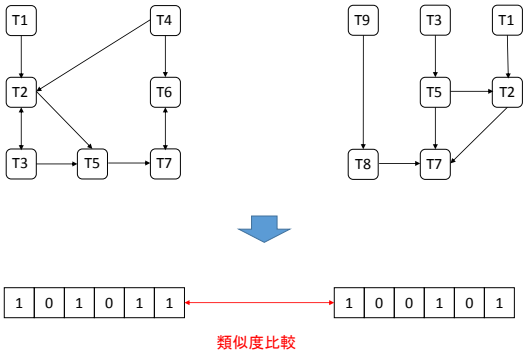


図 5　実際に映画のグラフの類似度比較した例．グラフ同士が部分的に類似しているため，分散表現が近似した．

表 1　使用したあらすじの文字数分布．1,000 文字以下のあらすじが全体の 8 割を占めている．

文字数	件数
151〜300 文字	2,753
301〜500 文字	1,926
501〜1,000 文字	1,938
1,001〜3,000 文字	1,547
3,001〜5,000 文字	64
5,001 文字以上	8

4.1　映画のあらすじからなるデータセットの作成

提案手法の評価のために，Yahoo!映画に登録されている映画のタイトル情報と Wikipedia の該当記事のあらすじ文を利用した．この際，あらすじ文が 150 文字以下の映画は取り除いた．これは，Wikipedia が有志によって編集されているため，必ずしも十分な情報が掲載されていないことを考慮したためである．使用した映画は 8,236 本，あらすじ文は最長で 7,875 文字，平均は約 650 文字だった．使用したあらすじ文の文字数の分布を表 1 に記す．

4.2　実　　装

Yahoo!映画のタイトル情報を用いて，Wikipedia の該当記事からあらすじ文を抽出した．抽出した映画のあらすじ文を CaboCha を用いて係り受け解析した．ノードに用いた単語の分散表現を Word2Vec を用いて算出した．算出した単語の分散表現を k-Nearest Neighbor（k-NN）を用いてクラスタリングした．クラスタリングによって学習したラベル情報を用いて，あらすじ文の単語表現を統一した．あらすじ文を NetworkX[1] を用いてグラフ化した．そのあとで，グラフの分散表現を graph2vec [9] を用いて算出した．グラフの分散表現の類似度算出にはコサイン類似度を用いた．Cabocha は，パラメータを指定せずデフォルトの設定のまま使用した．Word2Vec を使用する際には，算出する分散表現を 100 次元に設定した．k-NN を使用した際に，Word2Vec の類似度算出を用いて類似度が 0.8 を超える単語に同一のラベルを割り振った．映画のあらすじ文のコーパスの三分の一にラベルが割り振られた段階で，k-NN を使用して残りの単語にラベルを割り振った．また，k-NN で 2 つの近傍を確認するように設定した．graph2vec でグラフの分散表現を算出する際に，分散表現は 1,000 次元に設定し，学習は 1,000 回行った．

1：NetworkX:
https://networkx.org/

表 2　各手法で適用した処理の一覧

手法	ベースライン	グラフ化のみ	抽象化のみ	提案手法
グラフ化	×	○	×	○
抽象化	×	×	○	○

各手法について具体的には，

- **ベースライン**：映画のあらすじ文を Doc2Vec によるテキスト類似度によってランキング，
- **グラフ化のみ**：映画のあらすじ文をグラフ化し，グラフの分散表現の類似度を用いてランキング，
- **抽象化のみ**：映画のあらすじ文中の単語について表現の統一を行い， テキスト類似度によってランキング，
- **提案手法**：映画のあらすじ文中の単語について表現の統一を行い，あらすじ文をグラフ化し，グラフの分散表現の類似度によってランキング

するように実装した．それぞれのランキングの上位のアイテムについて，それぞれの手法の結果をランダムに混ぜた状態で被験者に見せ，類似度を判定した．表 2 に各手法で使う技術を簡潔に示す．

4.3　実 験 設 定

あらかじめデータセットに含まれる映画を，無作為に 12 件選出する．本研究では語の抽象化とグラフ化という 2 点の技術的な工夫があるので，それぞれを組み合わせた 4 つの手法と，ランダム手法の合計 5 つの手法を比較した．被験者は，12 件の映画について，映画ごとに映画のタイトルとあらすじが与えられる．次に，50 本の映画タイトルとあらすじについて，4 段階で類似度を判定する．40 本のうちわけは，4 つの手法の出力を無作為に並び替えたもので，それにランダムに選出した 10 本を加えて評価実験を行う．

4.4　実 験 手 順

被験者にクエリとして入力した映画と各手法で出力した映画のあらすじ文が記載された Google フォームの URL を配布した．被験者にクエリの映画のあらすじ文と各映画のあらすじ文の類似度を 4 段階で評価させる．各手法のスコアから，グラフ化と抽象化が有効であるかを判定する．

4.5　実 験 結 果

本研究では，各手法の出力は，任意の映画に対する類似度による映画のランキングである．そのため実験結果の評価には，情報検索における適合度とランキングの評価尺度である P@k（Precision@k）と nDCG（normalized Discounted Cu-

表 3　各手法の評価尺度の全クエリ平均. すべての手法でベースラインが最も高いことが分かる.

評価法	提案手法	抽象化のみ	グラフ化のみ	ベースライン	ランダム
P@1	0.00	0.33	0.00	**0.67**	0.00
P@5	0.02	0.27	0.02	**0.60**	0.00
P@10	0.04	0.24	0.03	**0.49**	0.02
nDCG	0.41	0.57	0.42	**0.71**	0.42

mulative Gain）をそれぞれ用いた.

P@k とは，ランキング内の上位 k 件のうち，適合する（すなわち，被験者からの評価値が閾値を超える）データが何件含まれているかで評価する評価尺度である．nDCG とは，実際のランキングが理想のランキングとどの程度共通しているかを，相対的に 0 から 1 の範囲で算出する評価尺度である.

実際の nDCG の計算として，初めにランキングごとに DCG を計算し，それを全タスクで正規化して合算して用いる．この際，DCG は，評価対象の数を k，順位を i，スコアを r とすると，

$$\mathrm{DCG} = r_1 + \sum_{i=2}^{k} \frac{r_i}{\log_2(i)} \tag{1}$$

として定義される．nDCG は，理想のランキングの DCG を $DCG_{perfect}$ とすると，

$$\mathrm{nDCG} = \frac{DCG}{DCG_{perfect}} \tag{2}$$

で算出する.

各手法の各評価法のスコアを表 3 で示す. 全ての評価手法でベースラインが最も高いスコアとなっている. 抽象化のみ行った手法が 2 番目に高いスコアとなっている. ランダム手法のスコアが低いことから, データセット内に正解データが少ないということが分かる. また, 提案手法とベースライン手法のクエリごとの各評価法のスコアを表 4 で示す. すべてのスコアが提案手法よりベースラインが高くなっている. nDCG と precision@10 のスコアに注目して, 最もスコアの高かった「容疑者 X の献身」の出力結果を表 5 で, 最もスコアの低かった「パイレーツ・オブ・カリビアン/最後の海賊」の出力結果を表 6 でそれぞれ示す. なお, 表 3, 4, 5, 6 中の映画のタイトルはスペースの都合で必要に応じて略称を記載している.

5 考　　察

本章では，被験者実験の各手法の結果を元に，使用した技術ごとの精度への影響と原因について議論する．全体を通して，提案手法は特に既存手法に対して，検索精度が低かった．原因として考えられることとして，物語のグラフ化に係り受け解析を用いた点，語の抽象化に k-NN を用いた点が挙げられる．また，そもそも，ベースライン手法からして，検索精度が低かった可能性がある．最後に，評価時の被験者への設問が適切でなく，一部の映画で正しくラベル付けされなかった可能性がある．これらの可能性について論じ，改善策について議論する.

本実験用の実装では，物語のグラフ化に係り受け解析結果を用いた．一方で，係り受け解析は，あくまで文章の構造をグラフ化するためのものであるので，物語展開そのものをグラフ化するには不十分であったことが考えられる．具体例として, 図 6 に「桃太郎は鬼退治した」という文章のグラフ化について示す. あらすじ文のグラフ化の際に, 名称やイベントがノードとなり, 関係や動作がエッジに情報として付与されているグラフが望ましいと考えられる. あらすじ文のグラフの分散表現のコサイン類似度を算出し比較した．今後，グラフエンベッディング的なアプローチで物語展開を比較するためには，既存の研究でクラウドソーシングで作成しているプロットグラフなどの，より高度なグラフ化手法が必要だと考えられる.

表 4　提案手法とベースライン手法に注目したクエリごとの各手法の精度の評価結果. ベースラインがすべての評価尺度において高性能なことが分かる.

	nDCG		P@1		P@5		P@10	
	提案	ベース	提案	ベース	提案	ベース	提案	ベース
ボヘミアン	0.41	0.74	0.00	1.00	0.00	1.00	0.10	0.50
ジュラシック	0.42	0.71	0.00	1.00	0.00	0.40	0.00	0.40
スター・ウォーズ	0.41	0.85	0.00	1.00	0.00	0.60	0.00	0.50
グレイテスト	0.42	0.71	0.00	1.00	0.20	0.80	0.10	0.50
万引き家族	0.45	0.68	0.00	0.00	0.00	0.20	0.10	0.40
ファンタビ	0.38	0.81	0.00	1.00	0.00	0.80	0.00	0.80
怪盗グルー	0.47	0.50	0.00	0.00	0.00	0.00	0.00	0.00
パイレーツ	0.32	0.73	0.00	1.00	0.00	0.80	0.00	0.80
SING/シング	0.38	0.62	0.00	0.00	0.00	0.20	0.00	0.30
ローグ・ワン	0.49	0.87	0.00	1.00	0.00	0.80	0.00	0.60
容疑者 X の献身	0.42	0.50	0.00	0.00	0.00	0.60	0.20	0.40
ハリー・ポッター	0.36	0.80	0.00	1.00	0.00	1.00	0.00	0.70

表 5　クエリ中で最も提案手法の評価尺度の高かった「容疑者 X の献身」の出力結果一覧

	提案手法		ベースライン	
順位	出力結果	評価値	出力結果	評価値
1	シンプル・プラン	2.00	県警対組織暴力	2.00
2	変人村	2.00	スマホを落とした	3.33
3	シャッター アイランド	2.67	魍魎の匣	2.33
4	童貞ペンギン	2.00	デビルズ・ノット	3.33
5	恋の時給	1.00	さらば愛しき女よ	3.00
6	勇者たちの戦場	2.00	あの頃, 君を追いかけた	1.66
7	ノ・ゾ・キ・ア・ナ	2.00	天使のいる図書館	1.33
8	君のまなざし	3.00	NIGHT HEAD	2.00
9	死者の学園祭	3.33	検察側の罪人	3.33
10	幸福への招待	1.33	マイアミ・バイス	2.00

表 6　クエリ中で最も提案手法の評価尺度の低かった「パイレーツ・オブ・カリビアン/最後の海賊」の出力結果一覧

	提案手法		ベースライン	
順位	出力結果	評価値	出力結果	評価値
1	愛と追憶の日々	1.33	ワールド・エンド	3.33
2	私の少女	1.00	呪われた海賊たち	3.33
3	大いなる幻影	1.33	闘将スパルタカス	1.67
4	パラサイト・イヴ	1.67	海底二万哩	3.33
5	青春の輝き	1.66	タンタンの冒険	3.00
6	マイ・ファニー・レディ	1.33	SF 巨大生物の島	3.00
7	スカイキャプテン	1.33	デッドマンズ・チェスト	3.33
8	ワールド・トレード・センター	1.33	ドラえもん	2.66
9	ハードロマンチッカー	1.00	ウォーターワールド	3.00
10	14 歳	1.33	シンバッド七回目の航海	3.33

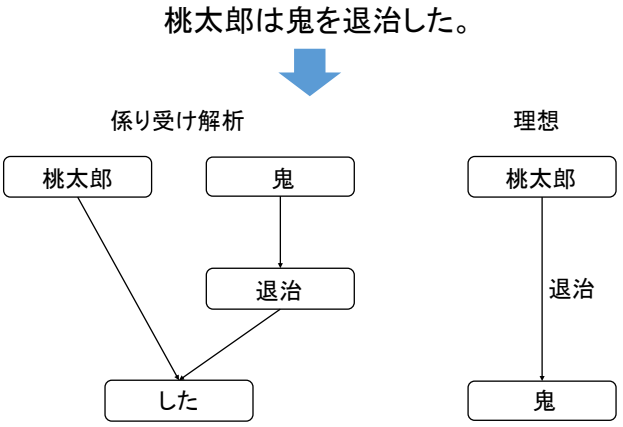


図 6　係り受け解析を用いたグラフ化と理想のグラフ化の例．名称やイベントがノード, 関係や動作がエッジとして構成されるグラフが望ましい．

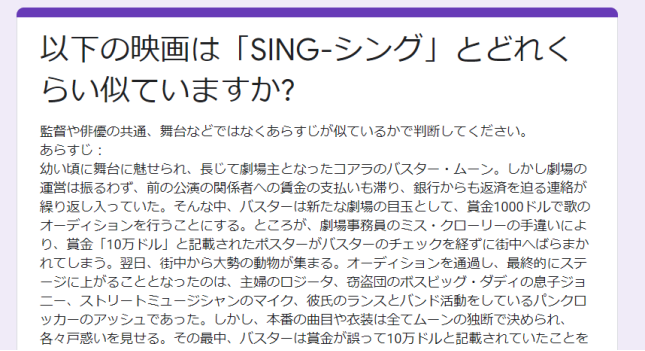
また，語の抽象化についても，実験に用いたあらすじの各語の抽象化について人手で確認すると，無関係な語同士が同じラベルに割り振られている場合が多かった．本研究では，計算コストの観点からクラスタリングに k-NN を用いたが，ノイズデータが含まれるデータの処理に弱い k-NN は，本研究のデータセットの処理に不適当であった．また，参照する近傍の数を初期値である 2 に設定したが，k-NN は多数決を行ってラベルを決定しているため，参照する近傍の数は奇数で設定するべきであった．本研究では，8,236 本の映画を対象に処理を行ったが，扱う映画の本数を減らし，DBSCAN 等でクラスタリングを行うことで精度が向上すると考えられる．

使用したデータセットそのものが原因で，提案手法，ベースライン両方の検索精度に影響を与えたことが考えられる．今回の実験では，Wikipedia の記事情報を人手で作成したルールに従って正規化していたが，正規化の精度が低く一部に不要な情報が含まれていた．これは，Wikipedia が有志によって自由に編集されるソーシャルサイトであるためで，具体的には，

- あらすじ文が映画の結末まで書かれているか，
- 監督や俳優の得た情報を含んでいるか，
- 文章の形式が整っているか，
- 正しい文法で書かれているか

などの観点で，記事ごとに記述量や品質の差が大きかった．このことから，Wikipedia のあらすじ文は差が大きく，今回のような自然言語処理を伴う実験には不向きなデータだった可能性が考えられる．

また，映画のあらすじ文全体で約 13 万種の単語が含まれていたが，そのうち約 7 万 8 千種の単語が 1 度しか登場しないユニークな単語であった．これにより，Doc2Vec による分散表現化の精度が下がったと考えられる．今回の実装では，あらかじめ係り受け解析を行って文章をノードに分割してから，ノードごとに分散表現化を行った．この際に，係り受け解析の結果で語同士が結合される場合があり，前処理による統一が不十分となったためである．


以下の映画は「SING-シング」とどれくらい似ていますか?

監督や俳優の共通、舞台などではなくあらすじが似ているかで判断してください。
あらすじ：
幼い頃に舞台に魅せられ、長じて劇場主となったコアラのバスター・ムーン。しかし劇場の運営は振るわず、前の公演の関係者への賃金の支払いも滞り、銀行からも返済を迫る連絡が繰り返し入っていた。そんな中、バスターは新たな劇場の目玉として、賞金1000ドルで歌のオーディションを行うことにする。ところが、劇場事務員のミス・クローリーの手違いにより、賞金「10万ドル」と記載されたポスターがバスターのチェックを経ずに街中へばらまかれてしまう。翌日、街中から大勢の動物が集まる。オーディションを通過し、最終的にステージに上がることとなったのは、主婦のロジータ、窃盗団のボスビッグ・ダディの息子ジョニー、ストリートミュージシャンのマイク、彼氏のランスとバンド活動をしているパンクロッカーのアッシュであった。しかし、本番の曲目や衣装は全てムーンの独断で決められ、各々戸惑いを見せる。その最中、バスターは賞金が誤って10万ドルと記載されていたことを

図 7　実験に用いたフォームの 1 例．注意事項が目に留まりにくい配置になっている．

また，ユニークな単語が多くなってしまったため，本研究の要である，固有名詞の抽象化が十分に行われなかった可能性がある．本研究では，前処理で「・」を含む単語を分割し，以降に登場する単語に置き換える処理を行ったが,「・」を含まない単語についても処理を行うことでユニークな単語が減少し，分散表現化の精度が向上すると考えられる．ユニークな単語の分散表現は，学習に用いる単語が前後の単語に限定されてしまい，意味の全く異なる前後の単語に類似した分散表現が算出されてしまった．

例えば,「シンドバッド虎の目大冒険」という映画のあらすじ文を解析する際に,「賢人メランシアス」,「賢者メランシアス」,「メランシアス」という，3 つのユニークな単語が登場した．このため，3 つの単語はそれぞれが単純に前後の単語と類似した分散表現となってしまった．ルールベースの前処理を徹底し，表記を「メランシアス」で統一することにより，学習に用いる単語を増やし，分散表現の精度を向上させられると考えられる．

評価タスクの設定において，ベースラインに比べて提案手法が不利になった可能性も考えられる．テキスト比較を用いたベースライン手法のスコアが高くなった要因として，シリーズ作品をクエリとして入力した際に，シリーズの他作品を類似度が高いと判定した被験者が多かったと考えられる．シリーズ作品はあらすじ文にシリーズ固有の単語が含まれることが多いため，ベースライン手法で出力されることが多かった．

本研究は，こういったメタデータによらない検索方法を提案しているが，シリーズ作品はあらすじ文の内容に関わらず高いスコアになる傾向が強く，メタデータの影響を排除しきれなかった．

その要因の 1 つとして，図 7 のように，質問の見出しとクエリとして入力した映画のあらすじ文の間に注意事項を配置したので，被験者の印象に残りにくかったと考えられる．

実際のデータを用いた被験者実験を通して，提案手法では，十分な精度で類似したあらすじを持つ映画を発見することができないと分かった．一方で，様々な課題点が見出されており，今後，前処理手法やグラフ化に用いるアルゴリズム，クラスタリング手法をより発展的なものにすることで，精度を改善できることが見込まれる．

6 ま と め

本研究では，単語の抽象化とグラフエンベッディングを用いた固有名詞やメタデータによらない，物語展開の類似した映画の検索手法を提案した．Yahoo!映画のタイトル情報を抽出し，Wikipedia の該当記事からあらすじ文を抽出した．あらすじ文の係り受け解析を CaboCha を用いて行った．単語の分散表現化を Word2Vec を用いて行い，k-NN を用いてクラスタリングした．NetworkX を用いてグラフ化し，graph2vec を用いて分散表現化した．ベースライン手法，グラフ化のみを行った手法抽象化のみを行った手法，提案手法の比較を被験者に出力結果をスコアリングさせることで行った．

実験結果から，Doc2Vec を用いたベースラインが最も優秀な結果を残しており，本研究では抽象化，グラフ化を提案したが，いずれも精度を向上させられないことが分かった．

一方で抽象化のみ行う手法はベースラインに次いで 2 番目に高い精度を持つことが分かった．あらすじ文の処理の改善とクラスタリング手法の変更により，精度向上が見込めると考えられる．グラフ化は，最も制度が低く，本実験では係り受け関係を用いてグラフ化を行ったが，根本的な手法の見直しが必要だと考えられる．また，実験の際に被験者にメタデータによらない評価をさせる工夫が必要であることが分かった．検索結果の精度を向上させ，実際に実用的なレベルであらすじの類似した映画を発見可能にするためには，今後，これらの問題をひとつずつ解決してゆく必要がある．

謝　　辞

本研究は JSPS 科研費 18K18161（代表:荘司慶行），18H03243（代表：田中克己）の助成を受けたものです．ここに記して謝意を表します．

文　　献

[1] Joël Plisson, Nada Lavrac, Dunja Mladenic, et al. A rule based approach to word lemmatization. In *Proceedings of IS*, Vol. 3, pp. 83–86, 2004.

[2] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, Vol. 26, pp. 3111–3119, 2013.

[3] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International conference on machine learning*, pp. 1188–1196, 2014.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[5] Yoshinori Hijikata, Hidenari Iwai, and Shogo Nishida. Context-based plot detection from online review comments for preventing spoilers. In *2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 57–65. IEEE, 2016.

[6] Balamurugan Anandan, Chris Clifton, Wei Jiang, Mummoorthy Murugesan, Pedro Pastrana-Camacho, and Luo Si. t-plausibility: Generalizing words to desensitize text. *Trans. Data Priv.*, Vol. 5, No. 3, pp. 505–534, 2012.

[7] Latanya Sweeney. Replacing personally-identifying information in medical records, the scrub system. In *Proceedings of the AMIA annual fall symposium*, p. 333. American Medical Informatics Association, 1996.

[8] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 855–864, 2016.

[9] Annamalai Narayanan, Mahinthan Chandramohan, Rajasekar Venkatesan, Lihui Chen, Yang Liu, and Shantanu Jaiswal. graph2vec: Learning distributed representations of graphs. *arXiv preprint arXiv:1707.05005*, 2017.

[10] Smriti Bhagat, Graham Cormode, and S Muthukrishnan. Node classification in social networks. In *Social network data analytics*, pp. 115–148. Springer, 2011.

[11] O-Joun Lee and Jason J Jung. Story embedding: Learning distributed representations of stories based on character networks. *Artificial Intelligence*, Vol. 281, p. 103235, 2020.

[12] Boyang Li, Stephen Lee-Urban, George Johnston, and Mark Riedl. Story generation with crowdsourced plot graphs. In *Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013.

[13] Momo Kyozuka and Keishi Tajima. Ranking methods for query relaxation in book search. In *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pp. 466–473. IEEE, 2018.

[14] Bo Xiong, Gunhee Kim, and Leonid Sigal. Storyline representation of egocentric videos with an applications to story-based search. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4525–4533, 2015.

[15] Seung-Bo Park, Kyeong-Jin Oh, and Geun-Sik Jo. Social network analysis in a movie using character-net. *Multimedia Tools and Applications*, Vol. 59, No. 2, pp. 601–627, 2012.

[16] Daniel Billsus and Michael J Pazzani. A hybrid user model for news story classification. In *Um99 user modeling*, pp. 99–108. Springer, 1999.

[17] Judith Solomon, GEORGE CAROL, and Annemieke De Jong. Children classified as controlling at age six: Evidence of disorganized representational strategies and aggression. *Development and psychopathology*, Vol. 7, pp. 447–463, 1995.