

# Twitterにおけるアニメのネタバレツイート判定手法の提案

田島 一樹<sup>†</sup> 中村 聡史<sup>‡</sup>

<sup>†</sup> <sup>‡</sup> 明治大学大学総合数理学部 〒164-8525 東京都中野区中野 4-21-1

E-mail: <sup>†</sup> ev30559@meiji.ac.jp, <sup>‡</sup> satoshi@snakamura.org

**あらまし** アニメなどテレビ番組を視聴しながら Twitter で感想などの情報発信することは一般的であり、リアルタイムな感想共有は視聴体験を高めることにつながっている。一方で、こうしたネタバレ情報は、未視聴のユーザにとって本来作品を通して体験するはずだった興奮や感動的な体験を無くしてしまう忌むべきものである。我々はこれまでの研究で、放送時間差によってネタバレに遭遇してしまうユーザの規模を明らかにし、ネタバレデータセットの構築およびアニメにおけるネタバレ分類を行った。本研究では複数のアニメコンテンツに対するネタバレデータセットを構築し、ネタバレ判定手法を提案するとともに、評価実験でネタバレの推定可能性を検証する。結果として、アニメのネタバレ判定にはツイートに含まれるアニメの登場人物名を一般的な語に置換し、かつ係り受け解析を用いて単語ベクトルを生成することが必須であることを明らかにした。

**キーワード** ネットバレル防止, ストーリーコンテンツ, 機械学習, Twitter

## 1.はじめに

TwitterやFacebookに代表されるソーシャルネットワークサービス (SNS) は、友人や知人などとの交流や情報収集の場として必要不可欠な存在となりつつある。こうしたSNSでは互いの近況を報告するだけでなく、思ったことや感じたことをリアルタイムで発信していく、他者と共有するということが日常的に行われている。ここで、ドラマやアニメで思ったことや感じたことをTwitterで他者と共有するため発信することも多く、番組の放送に応じてTwitter上が盛り上がることも多い。こうした番組に連動したツイート (140字以内のTwitterでの投稿) は、その番組を視聴している人にとっては楽しいものである。一方、その番組を視聴するのを楽しみにしているが、何らかの事情でリアルタイムに視聴できない人にとっては、そうしたツイートはネタバレにつながるため悩ましいものである。

ここでネタバレとは、物語を視聴することを通して本来得られたはずの興奮や感動的な体験を奪い去ってしまうものである。株式会社社会情報サービスが運営しているサイト [1] で行われたコンテンツにおけるネタバレに関するアンケートの結果、受けた人の内の約6割が映画などのネタバレに対して不満をもっているものとしている。こうした問題はSNSサイトにアクセスすること、アプリケーションを使わないことで回避することができる。しかし、SNSを遮断することは友人とのコミュニケーションを遮断するとともに、ニュースなどの情報への接触機会を減らしてしまうため、その対策方法としては現実的ではない。また、ネタバレされないためにはリアルタイムで視聴したら良いと考えられるが、仕事や学校の都合上、リアルタイムで視聴できないケースは多々ある。さらに、地域による放送時間の違いも問題の一つである。SNSが普及する

までであればこうした地域が離れた視聴者同士は、その物理的な距離によりコミュニケーションを取る機会は限られており問題となりにくかったが、SNS上でこうしたユーザ同士が繋がったに問題となっている。

我々はこれまでの研究でドラマやアニメの放送時間が地域ごとに異なるという点に注目し、放送時間差によってネタバレに遭遇してしまう可能性があるユーザの規模について調査し、視聴者の約7割がネタバレに遭遇してしまう可能性があることを明らかにした。また、ストーリーコンテンツにおいて人々が共通して致命的なネタバレと考える出来事について調査を実施し、アニメでは正体、生死、人物特徴、勝敗の4カテゴリに関する情報が特に問題となるネタバレであること、そしてSVMによる分類を行った結果、勝敗に関するネタバレの判定はしやすく、正体に関するネタバレの判定はしにくいことを明らかにした。

しかし、これまでの研究ではネタバレデータセットを構築する際に著者がデータに対してラベリングを行っており、再現性の点で問題があった。また、我々は放送中のアニメ番組に連動したツイートに含まれるネタバレでなく、ユーザの知人や友人が発言するジャンルが様でないツイートに含まれるネタバレを判定対象としているものであり、それをどの程度判定できるのか明らかにできていなかった。

そこで本研究では、複数のアニメコンテンツに対するネタバレデータセットを構築し、ツイートに対する事前処理と単語ベクトル生成手法を提案するとともに評価実験によってどの事前処理と単語ベクトル生成手法の組み合わせがTwitterでのネタバレ判定に有効であることを明らかにする。具体的には、1つのアニメ作品から4話分とバトル系、ミステリー系の2ジャンルに該当するアニメ作品をそれぞれ4つずつ選定し、各

アニメ番組に連動したツイートと一定期間内に投稿された全てのツイートから無作為に選定したツイートで同一作品データセットとジャンル分けデータセットを構築する。そして、ツイートに対して事前処理を行った上で単語ベクトルを作成し、SVMを用いたデータセットごとのネタバレツイートの判定精度の算出や判定しにくいネタバレツイートの特徴を分析することでネタバレ推定可能性を検証する。

## 2. 関連研究

ネタバレ防止を目的とした研究はこれまでもいくつかなされてきている。

中村ら[3]は、諸事情によってリアルタイムでスポーツの試合を見ることができないユーザがウェブページを閲覧している際、試合の内容に関するネタバレに遭遇してしまうことを問題とし、そうしたユーザのためにウェブページにおけるテキスト情報の曖昧化処理によってネタバレを防止する手法を提案している。しかし、その手法では事前に用意したネタバレ用正規表現辞書とのマッチングが前提であり、辞書をメンテナンスする手間があった。そこで、白鳥ら[4]はそういったサッカーのネタバレを大まかに直接的ネタバレと間接的ネタバレに分類し、正規表現のみでは判定できない間接的に試合結果が分かってしまうネタバレの判定可能性を示した。本研究はこうしたネタバレ判定のための分類器を機械的に構築することを目的としている。

Golbeckら[5]はアメリカの時差で地域ごとに放送時間が異なることによりTwitterでネタバレされてしまう事例を紹介しており、世界的にもネタバレは問題となっていることが分かる。この研究ではドラマやスポーツに関するワードが登録されているブラックリストを生成することによってテキストのネタバレを検知し、そのツイートをブロックするためのミュートボタンを実装している。Golbeckらは全てのネタバレを検知し、再現率100%を目指しているという点で我々と同じであるが、本研究ではストーリーコンテンツにおけるネタバレを分類および判定するという点で異なる。

Twitterクライアントを実装することによりネタバレ防止を行う手法として[3]や[5]がある。こうした研究ではハッシュタグ付きツイートから時間的にパースする単語を抽出し、その単語を含むツイートを非表示にすることでネタバレを防止している。しかし、パースする単語が必ずネタバレというわけではない。本研究では複数のアニメに対するネタバレデータセットを構築し、ストーリーコンテンツにおける本質的なネタバレの防止を目指しているという点で異なる。

池田ら[6]はAmazon.comや価格.comのようなオンラインショッピングサイトにおいて、ユーザが商品購入

の意思決定する際に参考にするレビュー文に小説、映画、ゲームなどのストーリーコンテンツに関するあらすじが含まれることを問題に挙げており、人名辞書と意見辞書を用いてあらすじを表す文と意見文をそれぞれ判定し、あらすじ部分のみを隠して表示するシステムの提案と実装を行っている。判定する文章がストーリーコンテンツに関するものという点では同じだが、オンラインショッピングサイトとTwitterでは、感嘆文や登場人物の発言の有無など投稿される文章の傾向が大きく異なると考えられる。

前田ら[7]は、ユーザがストーリーコンテンツのレビューを参考にする際にネタバレに遭遇することを問題とし、ストーリーコンテンツに対する短文形式のネタバレデータセットを構築することでネタバレに関する単語がストーリーコンテンツ内にどのように分布しているかを調査しており、コンテンツの文書からネタバレに関連した単語を判断する手法について検討している。我々はコンテンツの文書そのものではなくコンテンツに対するツイートからネタバレ分類器を構築し、ネタバレ判定を行うという点でアプローチが異なる。

田中ら[8]はニコニコ動画においてネタバレとなるコメントが動画視聴中に流れてきてしまうことを問題に挙げており、ルールベース手法と機械学習によりネタバレコメントを検知する手法を提案している。ここではネタバレとなる重要な単語を捉える際に単語バーストを利用している。田中らは動画共有サイト上の蓄積された動画コンテンツに対するネタバレを防ぐことを目指しているが、本研究はリアルタイムのイベントであるアニメに対するネタバレを判定するものである。

Leavittら[9]はストーリーコンテンツの閲覧中にネタバレを知ってしまうために否定的な感情が生まれると考え、作品を知る前からその作品の知識を得ることによって途中でネタバレをされても楽しみを損なわずに作品を楽しめるかどうかの実験を行っているが、本研究では人々が既にストーリーについての知識を持っていることを前提としているものである。

## 3. 判定手法

ここでは、アニメの放送に連動した実況のためのTwitterでの投稿（以降、番組連動ツイート）とユーザの知人や友人が発言するジャンルが一樣でないツイート（以降、一般ツイート）からSVMのための形態素解析と係り受け解析による単語ベクトル生成手法を提案する。

### 3.1 単語ベクトル生成手法

アニメのネタバレツイートには日常的に使用しない特徴的な語が含まれると考えられる。そこで、我々はネタバレ特有語を学習することでネタバレのツイー

トとネタバレでないツイートを区別する手法を提案する。具体的には形態素解析エンジンの Mecab を使用することでツイートを単語に分割し、得られた単語の中からネタバレが含まれる文章に特有であると考えられる名詞、動詞、形容詞、連体詞、副詞の5つの品詞を利用する。また、それぞれ得られた単語については原形を使用して学習する。この手法による単語ベクトル生成手順を図1に示す。

ツイート例

犯人の正体は主人公が駅前で会ったおじさんだった

犯人の/正体は/主人公が/駅前で/会った/おじさん/だった

生成した単語ベクトル

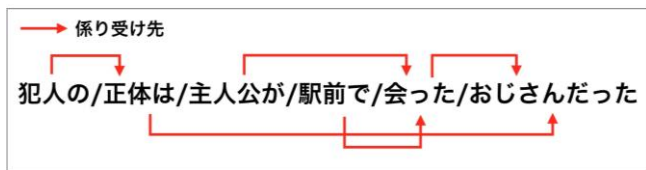
犯人, 正体, 主人公, 駅前, 会う, おじさん  
(名詞, 動詞, 形容詞, 連体詞, 副詞のみ)

図1 形態素解析を用いた単語ベクトル生成手順

形態素解析では単語の情報が得られるが、ネタバレツイートを判定する上で重要であると考えられる「誰がどうしたのか」などの文節同士の修飾関係を考慮することができない。そこで、比較のため係り受け解析を使用して文節同士の修飾関係を考慮することを可能にする手法を提案する。係り受け解析には日本語の係り受け解析器の CaboCha を使用する。この手法ではツイートを文節ごとに分割し、次に文節の係り受け先の情報を取得することで文節と修飾関係にある文節同士を繋げた文章で単語ベクトルを生成する。この手法による単語ベクトル生成手順を図2に示す。

ツイート例

犯人の正体は主人公が駅前で会ったおじさんだった



生成した単語ベクトル

- ・文節のみ  
犯人の, 正体は, 主人公が, 駅前で, 会った, おじさんだった
- ・文節同士の関係性を考慮  
犯人の正体は, 正体はおじさんだった, 主人公が会った, 駅前で会った, 会ったおじさんだった

図2 係り受け解析を用いた単語ベクトル生成手順

これら方法で単語ベクトルを生成したものをベースライン手法とする。

### 3.2 事前処理

判定精度向上のため、ツイートを単語または文節に分割した時点で行う事前処理を3つ提案する。

Brodyら[10]はTwitterに代表されるSNSで単語の一部を連続させるなどして変化させることによって投稿者の強い感情を表す語を検出するという手法を提案している。そこで、アニメ視聴者が衝撃的な展開に感情を左右されて用いると考えられる「wwwwww」「勝ったあああああ」のような連続した記号と単語の末尾の母音部分を正規表現で「w」や「勝ったあ」の形に変換する。これを正規化手法とする。これにより、語尾の伸ばした数ごとに単語ベクトルが区別されてしまうという問題を防止でき、判定精度を向上させることができると期待される。

また、ストーリーコンテンツにおける主人公やライバル、犯人や被害者などの登場人物名は、作品と話数ごとに大きく異なる。過去のアニメ作品におけるネタバレから、新しいアニメ作品のネタバレを推定するには、この人物名を一般化することが重要になる。そこで、物語の進行を左右するような影響力のある人物名を「主要人物」、物語への影響力の少ない人物名を「モブ」と置き換えを行う。これを人物名一般化手法とする。この手法によって、例えば「主要人物が死んだ」と「モブが死んだ」ものとを区別することができ、判定精度の向上が期待される。なお、「主要人物」と「モブ」の区別には番組情報が掲載されているサイト[11]とWikipediaを利用して行った。具体的には、番組情報が掲載されているサイトにおいて出演者として掲載されている人物名を「主要人物」と定義し、Wikipediaに掲載されている全ての人物名から先述した「主要人物」となる人物名以外を「モブ」と設定した。

最後に、先述した正規化手法と人物名一般化手法の2つを同時に行うものを、正規化・人物名一般化組み合わせ手法とする。

## 4 ネットバレーデータセット構築

ツイートデータを収集・整形し、複数のアニメ作品に対するネタバレデータセットを構築する。

### 4.1 ツイート収集

ここでは、番組連動ツイートと一般ツイートの収集方法を示す。

#### 4.1.1 番組連動ツイートの収集

番組連動ツイートの多くはアニメ作品ごとに特有な語を含むと考えられる。そこで、作品特有語を学習可能にするために1種類の作品から4話分を選定する。また、アニメ作品はジャンルによってどのような出来事がネタバレになるのか異なるという問題があるため、判定する作品のジャンルと同ジャンルの作品の番組連

動ツイートで学習する必要があると考えられる。そこで、バトル系とミステリー系に該当するアニメそれぞれ4作品から、1作品につき1話分を選定する。つまり、9作品から12話分の番組連動ツイートを収集する。

ここで、アニメ視聴者のすべての番組連動ツイートを収集するためには、その時間帯のすべてのツイートを収集、選別する必要があるため精度問題が生じる。また、フォローされている人のみにツイートを公開されている場合に、そうしたツイートを収集することは出来ない。そこで、アニメなどの作品を視聴しながらリアルタイムで投稿する際、ハッシュタグと呼ばれる検索およびタグ付けを可能とするキーワードをツイートに付与することが多いため、アニメ番組に対するハッシュタグ付きのツイートが番組連動ツイートを代表していると考え、学習および分類に使用する。

なお、Twitterでアニメ番組の実況に用いられるハッシュタグ(#シャロット、#tokyomxなど)を設定し、Twitter Search APIを利用して選定したアニメの番組連動ツイートを収集した。

#### 4.1.2 一般ツイートの収集

TwitterのStreamingAPIを利用し、日本語で投稿された全ツイートの中から無作為に5000件収集した。具体的には、データにネタバレツイートが極力含まれないように収集する時間帯を考慮し、東京の地域ではアニメ番組が放送されていなかった2015年1月9日16時から1時間ツイートを収集した。

#### 4.1.3 データ整形

収集した投稿の中には分類を行うデータとして不適切なものが含まれており、下記に示すパターンマッチによるテキスト処理を行った。

- (1) ボット(bot)と呼ばれる自動発言システムによる番組に連動した投稿の多くは、放送開始・終了等を知らせる広告であり、番組の内容について言及するものではなく不要である。そこで、「【自動】」または「【定期】」を含むツイートをボットの発言として除去した。
- (2) スпамツイートは番組に無関係のため不要である。そこで、スパムツイートに付与されがちな「http」を含むツイートを除去した。
- (3) リツイート(RT)と呼ばれる他人の発言を引用できる機能による投稿はテキストデータが重複するため不要である。そこで、「RT」を含む投稿をリツイートとして除去した。
- (4) 番組連動ツイートに含まれるネタバレとは無関係の文字列であるハッシュタグを「#」から改行までとして除去した。また、ハッシュタグが付与された一般ツイートは何かしらのイベントに連動して投稿された可能性があり、その中にはアニメ番組

に関するツイートも含まれるため、「#」を含む一般ツイートを除去した。

## 4.2 番組連動ツイート評価システム

データセット構築のため、収集したツイートがネタバレかどうか評価者を集めて分類を行ってもらった。ここでは、選定したアニメ番組に連動したツイートからそれぞれ2000または3000件ずつ無作為に抽出したツイートを対象とし、1話につき3人の評価者に分類を行ってもらった。なお、分類作業を行ってもらうために、図3に示すウェブシステムを開発した。このシステムでは、ユーザは最初にアカウント名を入力してログインし、ページ上に提示されているツイートに対してネタバレと感じるものを複数選択するというものである。ツイートは4.1.3項と同じテキスト処理を行ったものであり、投稿された時間順ではなくランダムに提示される。

The image shows a web application interface for evaluating tweets. At the top right, there is a link for 'Logout (ログアウトする)'. Below it, a message says '必ず確認してください' (Please confirm). There are two checked radio buttons: '判定するに当たって気をつけること' (Be careful when judging) and '操作方法' (Operation method). Below these are several radio button options for selecting categories: 'そんな、ばかな' (That's stupid), 'よくお気づきね' (You know that well), 'ノラガミかな' (Is it Norgami?), 'このコンゴウティーパーティー感' (This Congo Tea Party feeling), '自由そのものだ' (It's free as that), '自白が取れるまで' (Until the confession is taken), 'すべてがFになったの' (Everything became F), 'つちのなかにいる' (I'm in the ground), 'メンヘラの連続' (A series of menhela), '生き残り' (Survivor), and '悪い夢をみたな早く忘れたい' (I had a bad dream, I want to forget it quickly). At the bottom, there is a button labeled '次へ(残り7セット)' (Next (7 sets remaining)).

図3 開発したウェブシステム

評価者には作成したウェブサイトアクセスし、直前の話の内容を確認してもらった後にツイートの分類を行ってもらった。なお、ツイートに対する分類結果は100件毎にデータベースに記録されるため、途中で中断して再開することも可能となっている。評価者はTwitterを普段から用いており、かつ分類するアニメ作品の選定した話数まで視聴済みの20代の大学生の男性16名と女性3名である。

## 4.3 一般ツイートと番組連動ツイートで構成さ

## れたデータセット

一般的に、Twitterにおいてユーザの友人や知人の投稿内容は投稿ごとに言及するジャンルが異なることが多く、その中にネタバレが含まれてしまっているという状況を想定したデータセットを構築する必要がある。そこで、ネタバレ分類システムにおいて評価者3人のうち2人以上がネタバレと判定したツイートをネタバレツイート、4.1.2項で収集したツイートを非ネタバレツイートとして使用した。ネタバレツイートと非ネタバレツイートの学習量は偏りを無くすためにアンダーサンプリングを行った。ここでは、話数ごとにネタバレツイートと同数の非ネタバレツイートを無作為に選定した。

ここで、選定した1種類の作品のみを用いて構築したデータセットを同一作品データセットとし、この内容を表1に示す。ネタバレツイート率は分類した全ツイート中のネタバレツイートの割合で表される。

表1 同一作品データセット

作品名	ネタバレツイート率(%)
Charlotte(シャーロット) 第4話	7.7
Charlotte(シャーロット) 第7話	3.0
Charlotte(シャーロット) 第9話	17.3
Charlotte(シャーロット) 第13話	12.0

バトル系とミステリー系に該当するアニメ作品を用いて構築したデータセットをジャンル分けデータセットとし、この内容を表2に示す。

表2 ジャンル分けデータセット

ジャンル	作品名	ネタバレツイート率(%)
ミステリー系	六花の勇者 第12話	9.4
	すべてがFになる 第10話	6.7
	櫻子さんの足元には死体が埋まっている 第11話	1.2
	終物語 第5話	8.6
バトル系	Fate/stay night[Unlimited Blade Works] 第24話	12.7
	遊戯王 ARC-V 第82話	16.0
	ワンパンマン 第11話	4.7
	黒子のバスケ 第75話	11.3

## 5. 評価実験

ここでは一般ツイートに含まれるネタバレツイートの判定精度を算出する。

## 5.1 評価尺度

本研究ではクラスは2つあり、このクラスはネタバレとなるクラス(正例)と非ネタバレ(負例)である。この2クラスにおける評価尺度として、適合率(Precision)と再現率(Recall)を用いる。ここで、クラス $C_i$ に対して、適合率と再現率は以下のように算出される。

$$\text{Precision}(C_i) = \frac{\text{正しく } C_i \text{ に分類されたツイート数}}{C_i \text{ に分類されたツイート数}}$$

$$\text{Recall}(C_i) = \frac{\text{正しく } C_i \text{ に分類されたツイート数}}{C_i \text{ に属するツイート数}}$$

番組を楽しみに待っている視聴者にとってネタバレは可能な限り回避したいものである。正例であるネタバレツイートを可能な限り網羅することが重要である。そこで本研究では、番組が放送されるまでの期間中のみ、ある程度ネタバレでないツイートを遮断しても仕方ないものとし、正例(ネタバレ)分類に対する再現率に特に注目して評価を行う。

同一作品データセットでは過去の話数分のツイートを学習データとして利用し、新しい話数をテストデータとして適合率と再現率を算出する。ジャンル分けデータセットではジャンルごとに選定した4話の内3話を学習データとして利用し、残り1話分の作品をテストデータとして適合率と再現率を算出する。これを作品全てに対して算出し、適合率と再現率の平均を計算する。なお、機械学習におけるネタバレ判定精度については、3章で提案したベースライン手法、正規化手法、人物名一般化手法、正規化・人物名一般化組み合わせ手法でそれぞれ算出する。

## 5.2 結果と考察

評価実験による判定精度を図4~11に示す。なお、図4~7の横軸は判定した話数を表しており、図8~11の横軸は解析方法を表している。図4, 6, 8, 10の縦軸は適合率、図5, 7, 9, 11の縦軸は再現率を表している。

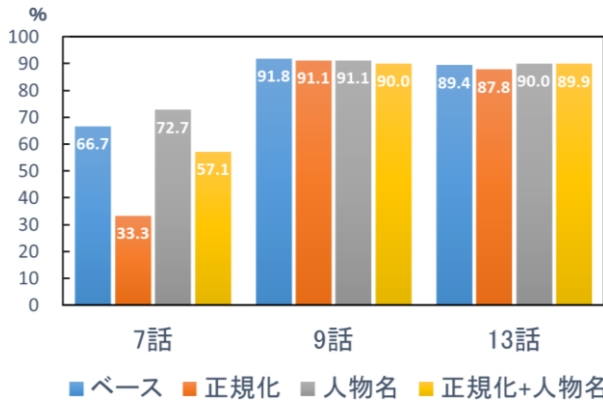


図4 同一作品の話数ごとの適合率（形態素解析）

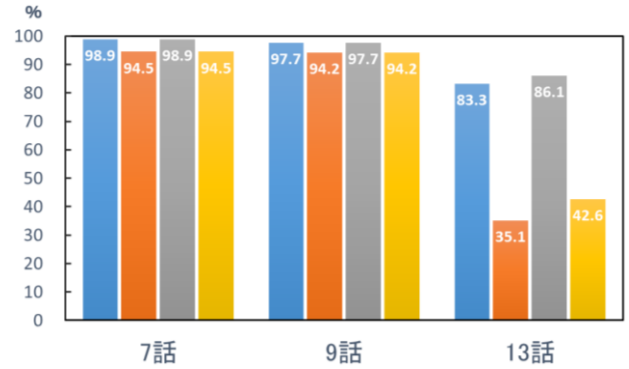


図7 同一作品の話数ごとの再現率（係り受け解析）

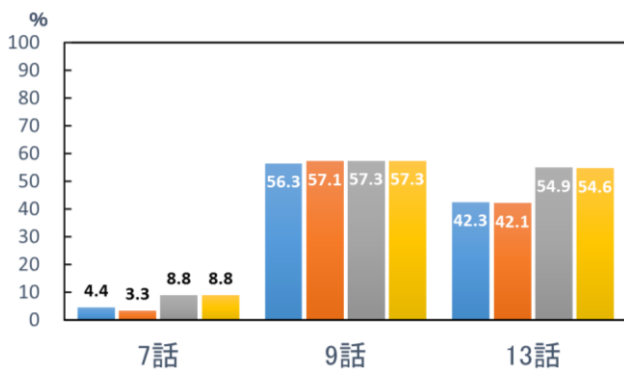


図5 同一作品の話数ごとの再現率（形態素解析）

図4, 5より形態素解析を用いたときの同一作品のネタバレ判定結果として、特に9, 13話の適合率が高く、ネタバレの誤検知は少ないが、一方で7話の再現率は10%未満、9, 13話でも再現率60%未満であり、あまりネタバレを網羅できなかったことが分かる。

手法ごとの結果では、人物名一般化手法で再現率が全て改善されている一方で正規化手法では7, 13話の判定精度が下がった。

図6, 7より係り受け解析を用いてネタバレ判定をした結果、形態素解析と比べて全体的に再現率がかなり高くなるのが分かる。特に7話は再現率がほぼ100%であり、ネタバレツイートをほとんど網羅することが可能であった。しかし、先の話数に進むにつれて少しずつ再現率が下がっている。一方、適合率は60%未満であり誤検知は増えた。

また、手法ごとの結果ではベースライン手法に比べて他の手法でほとんど判定精度が改善しなかった。

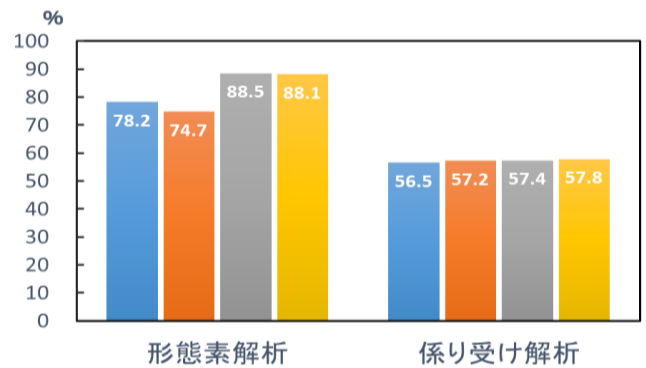


図8 バトル系における手法ごとの適合率

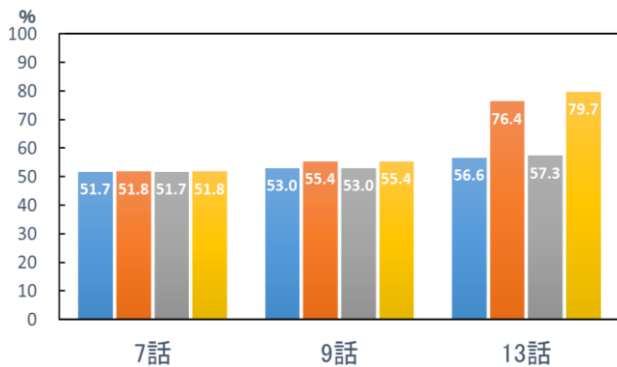


図6 同一作品の話数ごとの適合率（係り受け解析）

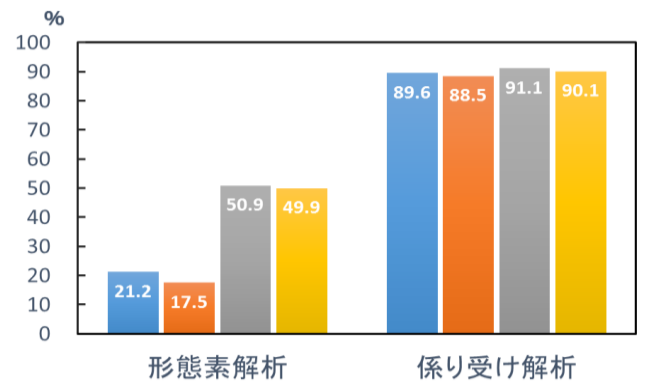


図9 バトル系における手法ごとの再現率

図8, 9より、係り受け解析を用いたときに形態素解

析と比較すると再現率がかなり高く、一方で適合率が低いことが分かる。また人物名一般化手法では形態素解析と係り受け解析の両方で判定精度の改善が見られ、特に形態素解析では適合率約 10%、再現率約 30%改善した。一方、正規化手法では改善しなかった。

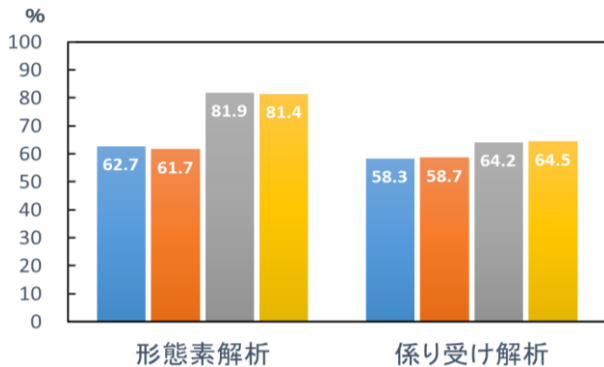


図10 ミステリー系における手法ごとの適合率

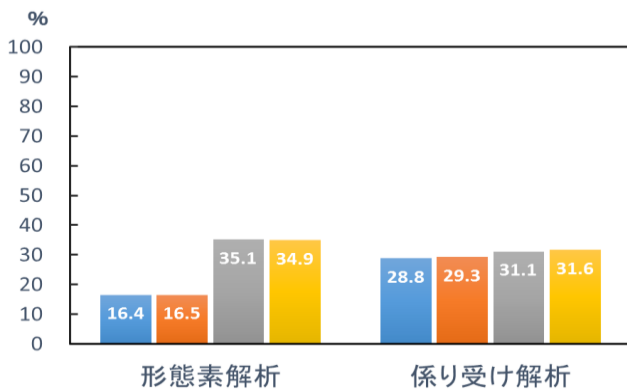


図11 ミステリー系における手法ごとの再現率

図 10, 11 より、全体的にかなり再現率が低く、ミステリー系のネタバレはあまり網羅できなかった。しかし、人物名一般化手法では特に形態素解析で適合率が約 20%、再現率が約 19%改善され、係り受け解析よりも判定精度が向上した。

ここで全体的な結果として、まず形態素解析と係り受け解析を用いた判定結果を比較したところ、基本的には係り受け解析を用いることでネタバレ判定をしやすくなることが明らかになった。しかし、ミステリー系においては形態素解析をしたときの判定精度が比較的やや高かった。これはミステリー系においては文節ごとの関係性よりも単語の情報が重要であったためと考えられる。そこで、アニメジャンルによって用いる単語ベクトル生成手法を変える必要があると考えられる。

次に、データセットごとの結果をまとめると同一作品データセットでは作品に特有な語を学習でき、かな

りネタバレを判定しやすいことが明らかになった。しかし、アニメコンテンツでは作品の話数が進むと展開が異なってくるため、有効な形態素が話数ごとに異なる。例えば、コミカルな場面からシリアスな場面に変化するなどの急展開後のネタバレ判定が困難であると考えられる。そこで、同一作品の最新話のネタバレを判定するときは過去に投稿されたその作品全ての番組連動ツイートを利用してデータセット構築するのではなく、最新話から数話前までの番組連動ツイートのみを利用する必要があると考えられる。ジャンル分けデータセットではバトル系のネタバレ判定はしやすく、ミステリー系のネタバレ判定は困難であることが明らかになった。その理由として、バトル系では「勝った」、「負けた」といった勝敗に関する語や「死んだ」、「生きていた」といった人物の生死に関する語が多用され、それらが判定に有効な形態素となり学習がしやすかったと考えられる。一方、ミステリー系では作品ごとに異なるトリックに関する情報や作品に特有な語が特に多く出現し、判定に有効な形態素が少なかったため学習がしにくかったと考えられる。

最後に手法ごとの結果をまとめると、正規化手法ではほぼ判定精度の改善は見込めないが、人物名一般化手法では判定精度を改善可能であることが明らかになった。また、正規化手法が有効でなかったため正規化・人物名一般化手法も人物名一般化手法と比べて有効でなかった。正規化手法については正例、負例のどちらにおいても感情を表現する連続語がほぼ等しく出現したため判定精度が向上せず、人物名一般化手法についてはネタバレツイートのみ一般化された登場人物名が多く含まれていたため判定に有効な単語ベクトルを生成でき、判定精度が向上したと考えられる。

ここで、実際に機械学習を通して出力されたデータを確認したところ、ネタバレと判定出来なかったツイートにはある程度傾向があることが明らかになった。そこで、判定が困難なネタバレツイートの特徴と言及されていた内容をデータセットごとに示す。

- 同一作品データセット
  1. 登場人物の正体に関する情報。
  2. 登場人物の見た目や内面といった特徴の変化が分かる情報。
- ジャンル分けデータセット
  - バトル系
    1. 登場人物の見た目や内面といった特徴の変化が分かる情報。
    2. ユーザの感情表現が含まれている。
  - ミステリー系
    1. 登場人物の正体に関する情報。
    2. トリックに関する情報。

