

# Paragraph Vectorに基づく感情極性を用いた Web 検索スニペットの生成方法

A Method of Generating Web Search Snippets Using Sentiment Polarity based on Paragraph Vector

寺澤 友治朗\*<sup>1</sup> 白松 俊\*<sup>1</sup> 大冨 忠親\*<sup>1</sup> 新谷 虎松\*<sup>1</sup>  
 Yujiro Terazawa Shun Shiramatsu Tadachika Ozono Toramatsu Shintani

\*<sup>1</sup>名古屋工業大学大学院情報工学専攻

Department of Computer Science and Engineering, Graduate School of Engineering, Nagoya Institute of Technology

Although the sentiment polarity is important for understanding public reputations, conventional web search snippets do not include the sentiment polarity. Hence, we implemented a system for generating web search snippets consisting of sentiment polarities. Our method is based on Paragraph Vector and logistic regression. Moreover, we evaluated the accuracy of our method using a corpus.

## 1. はじめに

インターネットが普及するにつれて、検索エンジンが多く利用されるようになった。検索エンジンは、ユーザの求めている情報をウェブ文書から探し、ユーザの求めている情報を含むウェブ文書をユーザに提供する。また、検索エンジンはウェブ文書に関する情報も提供する。特に、ウェブ文書に関する情報には検索結果スニペットが含まれる。検索結果スニペットとは、検索エンジンの検索結果に示されるウェブ文書の概要のことである。ここでのスニペットは、検索キーワードやその類語を含む文とその前後の文により構成される。このように、検索エンジンには、情報を提供する工夫がなされ、ウェブを利用する際には欠かせないものとなっている。

しかし、従来の検索エンジンには、評判情報の閲覧を支援する機能を備えていない。評判情報には見るべき情報を含む傾向がある。そのため、本研究では、検索エンジンに評判情報の閲覧を支援する機能を付加することを検討した。具体的には、Paragraph Vector を用いた感情極性分析を用いて、得られた肯定的な文と否定的な文を分けて、それらの文より検索結果スニペットを生成するシステムを開発した。

感情極性分析とは、特定の事物について述べた文が、当該事物に肯定的か否定的かを推定する処理である。このことから、感情極性分析は、評判分析の要素技術として、社会的需要が大きい。感情極性分析の活用例として、感情極性分析の行えるテキストマイニング API である TrueText\*<sup>1</sup> により、企業の評判や口コミの解析によって市場や会社の評判を即座に把握したり、製品や施設の利用者の要望および意見を集約できる。これらの機能によって、評判や要望、意見を容易に確認することができる。このように、評判情報は人々に重要な情報を与える。

## 2. 関連研究

本研究では、Deep Learning を用いた、自然言語処理技術を用いた。具体的には、Le ら [1] の提案した、Paragraph Vector (PV) を用いた。PV とは、自然言語の文よりベクトルを生成する、教師なしアルゴリズムである。PV は、Distributed Memory Model of Paragraph Vectors (PV-DM) と Distributed Bag of Words version of Paragraph Vector (以下 PV-DBOW) の 2 種類に分類される。

連絡先: 寺澤 友治朗, 名古屋工業大学大学院情報工学専攻,  
 tyujiro@toralab.org

\*1 <http://www.textmining.jp/>

PV-DM と PV-DBOW は、いずれも文や単語を入力を行うことで、文や単語の性質を示すベクトルを出力する。具体的には、単語の性質を示すベクトルは、文中における単語同士の共起により生成され、文の性質を示すベクトルは、文中の単語の並びより生成される。また、いずれの手法でも、可変長の文を固定長のベクトルにすることができ、また、入力の数が一定である機械学習に応用できる。そのため、PV による感情極性分析は、Bag-of-Words や n-gram モデルによる分析よりも有利である。

また、本研究で行う感情極性分析は、様々な応用方法が提案されてきた。近年では、ウェブサービスに対する研究もなされている。例えば、O'Connor ら [2] は、Twitter\*<sup>2</sup> というウェブサービスに投稿された公衆の意見に対して、感情極性を分析する研究を行った。具体的には、ある日付  $t$  に対してスコアリングを行った。“obama”, “mccain” など特定の単語を含む日付  $t$  の投稿に対して、スコアを考慮して、肯定的あるいは否定的な単語を含む投稿数を数える。肯定的な単語を含む投稿と、否定的な単語を含む投稿の数に応じて、日付  $t$  のスコアリングを行った。また、複数の日付  $t, t-1, \dots, t-j$  のスコアを考慮することで、月ごとのスコアリングができる。以上の手法により、“obama”, “mccain” の話題において、公衆の評判を明確にした。

Twitter のようなウェブサービスにおける、メタデータを利用した感情極性分析の研究もなされている。例えば、Wang ら [3] は、Twitter におけるハッシュタグと呼ばれるメタデータを利用して、感情極性を利用した。ハッシュタグとは、Twitter の投稿内に含まれる、語頭に“#”のついた単語のことである。ハッシュタグは、特定の話題を示す。つまり、同じ単語のハッシュタグを含む投稿同士は、同じ話題の投稿同士であることを示す。Wang らは、Twitter の投稿内の複数のハッシュタグの共起より、グラフィカルモデルを形成し、ハッシュタグに対する感情極性分析を行うアルゴリズムを考案した。

また、Kucuktunc ら [4] は、Yahoo Answers\*<sup>3</sup> と呼ばれるウェブサービスに対して感情極性分析を行った。Yahoo Answers は、質問および質問への回答が投稿ができる、スレッドフロー型の掲示板となっている。また、Yahoo Answers では、ユーザの住所や性別などの個人情報、投稿した質問のトピックなど、メタデータが含まれている。Kucuktunc らは、質問の投稿と回答の投稿に分けてそれぞれスコアリングを行い、ユーザや投稿した記事の情報より感情極性の情報を解析した。具体的には、投稿

\*2 <https://twitter.com/>

\*3 <https://answers.yahoo.com/>

した記事に対して、識別器を用いた感情極性分析を行い、記事に対するスコアを得る。得た記事のスコアを用いて、ユーザの投稿した質問のスコアの平均や、特定の質問に対する回答のスコアの平均などを算出した。他にも、特定のトピックに対するスコアの平均、特定の特徴を持ったユーザの質問のスコアの平均など、投稿内容やメタデータを利用した感情極性分析を行った。Kucuktuncらは、この手法により、投稿者や投稿内容に感情極性の傾向を調べた。

本研究では、検索エンジンに出力された文書を対象とした感情極性分析を行うことで、感情極性を考慮したスニペットの生成を行う。生成したスニペットは、ウェブ文書中の、肯定的もしくは否定的な文より構成される。そのため、文に対する感情極性分析をする必要がある。また、文に対する感情極性分析には、文中の単語の順序を考慮する必要がある。しかし、PVを用いていない、感情極性分析の先行研究では、単語に対する感情極性に着目している傾向がある。先行研究では、文に対する感情極性を行う際も、単語の語順が考慮されていない。一方、PVを用いれば、単語の数に関わらず、単語の語順すべてを考慮して、文を表現するベクトルを生成することができる。そのため、PVを用いた感情極性分析は、単語数に依存するn-gramモデルを用いるよりも、有効である。これらのことから、本研究では、PVを応用することで、感情極性を考慮したスニペットの生成を行った。

### 3. Paragraph Vector を用いた感情極性分析

本研究では、PVを感情極性分析に用いた。本研究では、PVにより生成した文の性質を示すベクトル(文ベクトル)を、多重ロジスティック回帰分析(以下MLRA)の素性を用いた。このようなMLRAを行うことで、ウェブ文書中の文の感情極性を分析する。また、本研究で用いたモデルの学習には、株式会社楽天の提供する楽天公開データ<sup>\*4</sup>を用いた。楽天公開データは、株式会社楽天が提供するウェブサービスのデータより構成されたデータである。楽天公開データに含まれるコーパスには、商品や旅行の情報およびレビューデータや、旅行のレビューデータに対して極性ラベルを付加した、TSUKUBAコーパスと呼ばれるデータセットなどが含まれている。TSUKUBAコーパスにおける極性ラベルは、3.2節で説明する。

#### 3.1 文ベクトルの生成

文ベクトルの生成には、PV-DMおよびPV-DBOWを用いた。PV-DMおよびPV-DBOWは、文 $s_1, s_2, \dots, s_n$ とその単語列 $w_{1,1}, w_{1,2}, \dots, w_{1,m}, w_{2,1}, w_{2,2}, \dots, w_{n,m}$ を用いて学習を行う。PV-DMおよびPV-DBOWにおける学習は、文 $s_1, s_2, \dots, s_n$ および単語列 $w_{1,1}, w_{1,2}, \dots, w_{1,m}$ を入力することで、行うことができる。また、学習の際には、文 $s_1, s_2, \dots, s_n$ および単語列 $w_{1,1}, w_{1,2}, \dots, w_{1,m}$ を入力することで、文 $s_1, s_2, \dots, s_n$ の文ベクトルを生成することができる。ここで、PV-DMによって生成した文ベクトルを $v_1^D, v_2^D, \dots, v_n^D$ とし、PV-DBOWによって生成した文ベクトルを $v_1^B, v_2^B, \dots, v_n^B$ とおく。生成した文ベクトルは $v_i = [v_i^D v_i^B]$ ( $i = 1, 2, \dots, n$ )とし、連結させる。連結させた文ベクトル $v_1, v_2, \dots, v_n$ は、文ベクトルとして扱い、感情極性分析の素性として用いる。また、学習後も、PV-DMおよびPV-DBOWのモデルを扱うことができる。学習済みのモデルを扱うことで、同様の手続きで文ベクトルの生成を行うことができる。本研究では、3.節で述べた楽天公開データよりPV-DMおよびPV-DBOWのモデルの学習を行い、学習済みのモデルより文書における文の感情極性の分析を行った。

\*4 <https://alaginrc.nict.go.jp/resources/rakutesn-dataset.html>

#### 3.2 文の感情極性の推定方法

文書の感情極性の推定を行う際には、文の感情極性の推定をする必要がある。そのためには、文の感情極性を識別する識別器の学習が必要となる。本研究では、3.節で述べた楽天公開データより、文ベクトルを生成し、文ベクトルを素性としたMLRAを行った。素性に対するラベルは、楽天公開データ中のTSUKUBAコーパスの極性ラベルを用いた。

本研究では、3.節で述べた楽天公開データを使用して、PV-DMおよびPV-DBOWのモデルの学習を行い、TSUKUBAコーパスの文より生成した文ベクトルを用いて感情極性分析のモデルを学習した。楽天公開データより学習に用いる文 $s_1, s_2, \dots, s_n$ より、3.1節で述べた方法を用いて、文ベクトル $v_1, v_2, \dots, v_n$ を推定した後は、MLRAを行う。本研究では、MLRAモデルの出力を、1,0,-1とした。ただし、1のラベルが付く文は、肯定的な文であり、-1のラベルが付く文は、否定的な文であり、0のラベルが付く文はいずれでもない文であるとする。学習の際は、TSUKUBAコーパスの文の文ベクトル $v_1, v_2, \dots, v_n$ と、対応する極性ラベルをデータセットとした。TSUKUBAコーパスでは、評価ラベルが文に付与されている。本研究では、“褒め”、“苦情”、“要求”、“ニュートラル”の4種類の評価ラベルのいずれかが付与されている文を用いた。ただし、学習の際は、ラベル“褒め”を1、ラベル“要求”、“苦情”を-1、ラベル“ニュートラル”を0と置き換える。以上の方法による学習によって、MLRAモデルの識別器を作成する。

#### 4. 感情極性を考慮したスニペット

本研究では、感情極性を考慮したスニペットおよびウェブ文書の感情極性の生成し、提示を行う。生成の際は、感情極性分析を用いた。また、提示を行う際には、ユーザの入力した検索質問と、ウェブ文書中の文の提示する必要性を考慮する必要がある。そのためには、生成したベクトルを用いる必要がある。

まず、検索エンジンに出力されるウェブ文書の文に対して、学習済みのモデルを用いた感情極性分析を行う。分析した文のうち、肯定的および否定的と判断された文に対して、以下の式を用いてスコアリングを行った。

$$score(s) = \lambda \max_u sim(s, u) + (1 - \lambda) \max_{w, q} sim(w, q) \quad (1)$$

ここで、 $\lambda$ は $0 \leq \lambda \leq 1$ とし、 $s$ はスコアリングの対象となる文、 $u$ は評価を示す文である。ここで、評価文 $u$ には、楽天公開データのTSUKUBAコーパスを用いた。また、 $w$ は、文 $s$ の文中の単語であり、 $q$ はユーザの入力した検索質問を示す。また、関数 $sim$ は、次の式で示される。

$$sim(p, q) = v_p \cdot v_q \quad (2)$$

ただし、 $v_p$ および $v_q$ は、それぞれ単語もしくは文 $p$ および単語もしくは文 $q$ より生成されたベクトルである。式1について、 $s$ との類似度が高い文 $u$ が存在するときや、文 $s$ 中に検索質問 $q$ と類似する単語 $w$ を含むとき、 $score(s)$ の値が高くなる。 $score(s)$ の値が高いものよりスニペットを生成することで、ユーザの入力した検索質問と、ウェブ文書中の文の提示する必要性を考慮したスニペットを生成できる。

式1を用いて、感情極性を考慮したスニペットを作成するアルゴリズムをAlgorithm 1に示す。感情極性を考慮したスニペットは、文書中の文 $s_1, s_2, \dots, s_n$ を入力することで、生成することができる。Algorithm 1では、 $s_1, s_2, \dots, s_n$ を $\mathbf{s}$ として扱う。Algorithm 1の2行目に示すように、入力した文 $\mathbf{s}$ より、

**Algorithm 1** Make Snippets

- 1: **Input:**  $s_1, s_2, \dots, s_n$  as  $s$
- 2: Generate  $v_{s_1}, v_{s_2}, \dots, v_{s_n}$  from  $s$
- 3:  $s \leftarrow \text{DescendingSort}(s, Eq.1)$
- 4:  $s^{pos}, s^{neg} \leftarrow \text{filter}(s, P(v_s) = 1), \text{filter}(s, P(v_s) = -1)$
- 5: **if** either  $s^{pos}$  or  $s^{neg}$  don't exist **then**
- 6:     **Output:** top three sentences by another
- 7: **else**
- 8:     **Output:**  $s_1^{pos}, s_1^{neg}, \text{MaxByScore}(s_2^{pos}, s_2^{neg})$
- 9: **end if**

ベクトル  $v_{s_1}, v_{s_2}, \dots, v_{s_n}$  を生成する。文ベクトルを取得したら、3 行目に示すように、式 1 に基づいた降順ソートを行う。ソート後は、4 行目に示すように、肯定的な文の集合および否定的な文の集合を取得する。ただし、 $P(v_s)$  は文  $s$  より推定した極性ラベルとする。肯定的な文の集合は  $s^{pos}$  とし、否定的な文の集合は  $s^{neg}$  とする。 $s^{pos}$  および  $s^{neg}$  はスニペットの生成に用いる。5 行目から 9 行目に示すように、 $s^{pos}$  もしくは  $s^{neg}$  が存在しないとき、他方より上位 3 文からスニペットを生成する。また、いずれも存在するときは、 $s^{pos}$  および  $s^{neg}$  の最もスコアの高い文 1 文ずつと、 $s^{pos}$  および  $s^{neg}$  の 2 番目にスコアの高い文より、スコアの高い方の文 1 文よりスニペットを生成する。ここで、8 行目に示す  $\text{MaxByScore}$  関数は、式 1 に示されるスコアの高い文を出力する関数である。

また、本研究では、感情極性を考慮したスニペットに加えて、ウェブ文書の感情極性も提示する。本研究では、ウェブ文書中の文を利用した。具体的には、4. 節に記した Algorithm 1 の、4 行目の手続きによって生成した、肯定的な文の集合  $s^{pos}$  および否定的な文の集合  $s^{neg}$  を用いて、ウェブ文書の感情極性の推定を行う。ウェブ文書の感情極性は、肯定的な文および否定的な文の全体に対して、肯定的な感情を含む文と判定された文の割合と、否定的な感情を含む文と判定された文の割合とした。このように、肯定的および否定的な感情を含む文の割合をウェブ文書の感情極性として示すことで、ウェブ文書に含まれる感情極性の概略を示すことができる。

**5. システム構成**

図 1 に本システムの構成図を示す。本システムの入力は検索質問であり、出力はスニペット列となる。具体的には、入力した検索質問を用いて、検索エンジン Google の検索結果を取得し、生成したスニペット列を適用させた検索結果を出力する。本システムでは、次の 5 ステップにより感情極性を考慮したスニペット列を生成する。ステップ 1 では、ユーザが入力した検索質問より、検索エンジン Google の検索を行う。ステップ 2 では、Google の検索結果よりウェブ文書を取得する。ステップ 3 では、取得したウェブ文書中の文に対して、感情極性分析を行う。ステップ 4 では、感情極性分析の結果を考慮して、Google 検索結果を修正する。検索結果の修正を行った後のステップ 5 では、修正した検索結果をユーザに示す。

本システムでは、まず、クライアントサイドにて、ユーザの入力した検索質問を受け取る。図 1 の“検索結果の取得”では、受け取った検索質問より、Google による検索を行い、検索結果を取得する。その後、取得した検索結果より、HTML のソースコードを解析し、検索結果に出力されたウェブ文書を取得する。具体的には、検索結果に出力されたウェブ文書の URL を取得し、取得した URL よりウェブ文書を取得する。

図 1 の“感情極性分析”では、取得したウェブ文書中の文よ

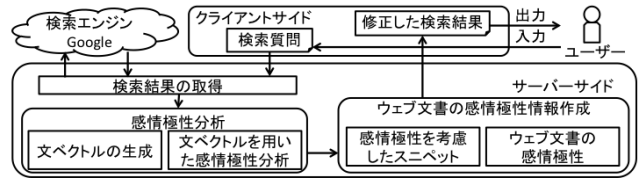


図 1: システム構成図

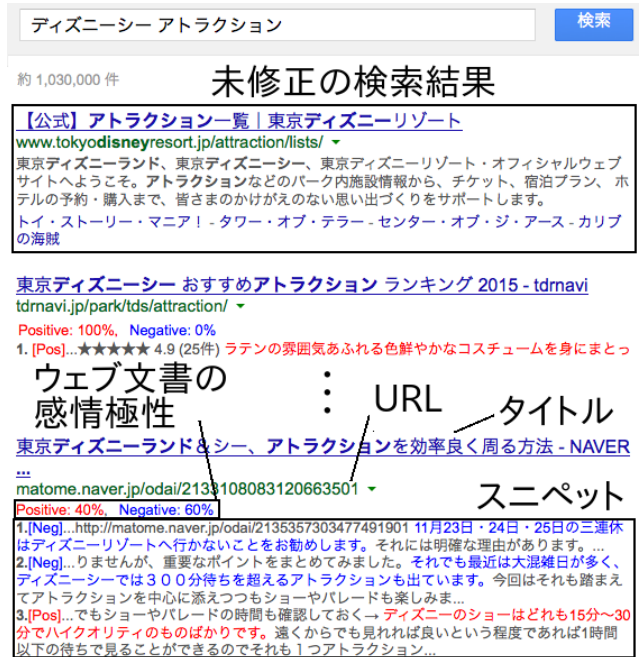


図 2: 本システムの出力結果

り感情極性分析を行う。感情極性分析は、図 1 に示すとおり、文ベクトルの生成と、文ベクトルを用いた感情極性分析の二つの段階に分けて行う。具体的には、ウェブ文書の文より文ベクトルを生成し、生成した文ベクトルを用いて感情極性分析を行う。文ベクトルの生成には 3.1 節で述べた手続きを行う。なお、文ベクトルの生成には、学習済みの PV-DM および PV-DBOW のモデルを用いる。感情極性分析には 3.2 節で述べた方法で学習した MLRA モデルを用いる。図 1 の“ウェブ文書の感情極性情報作成”では、ウェブ文書中の文と文の感情極性を用いて、感情極性を考慮したスニペットを作成し、ウェブ文書の感情極性を推定する。これらの手続きは、4. 節の方法で行った。以上の手続きを行った後は、検索結果を修正する。修正した検索結果は、クライアントサイドにて、ユーザに示す。

本システムの実行例を図 2 に示す。図 2 は、本システムのユーザが検索質問を入力した際に出力される画面となる。ただし、検索質問は、“ディズニーシー アトラクション”とした。図 2 の出力画面では、従来の検索エンジン Google の検索結果を一部修正したものとなる。また、図 2 では、実際の出力結果の一部を省略したものとなる。(省略は“…”で示す。) 図 2 の上部には、検索バーが表示され、その下には検索結果が表示される。検索は、図 2 の上部の入力欄で検索質問を入力して、検索ボタンをクリックすることで、行うことができる。ウェブ文書の情報を修正する際は、URL の下にウェブ文書の感情極性を示し、通常の見れば良いという程度であれば 1 時間



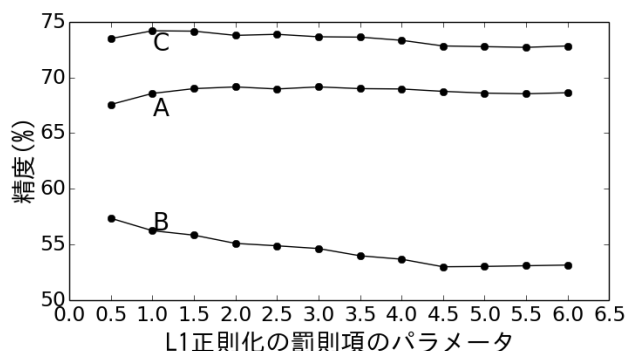


図 3: 評価実験の結果

ニペットへと変更する。

次に、修正したウェブ文書の情報の例について説明する。図 2 で“ウェブ文書の感情極性”と示される箇所は、ウェブ文書の感情極性の割合を示す。また、“スニペット”と示されている箇所は、感情極性を考慮したスニペットを示す。感情極性を考慮したスニペットは 4. 節で述べた方法により生成したものである。ただし、スニペット中の文で、肯定的な文は文字色が赤色、否定的な文は文字色が青色となる。これにより、スニペット中の文の感情極性を可視化する。また、肯定的および否定的な文の前後には、文書中における、肯定および否定的な文の前後の文の一部を付与する。なお、図 2 で、“未修正の検索結果”と示されている箇所は、4. 節の方法により、文書中の文の極性ラベルがすべて 0 と判断されたウェブ文書の検索結果である。このような検索結果の修正は行わないものとする。

## 6. 評価実験・考察

本研究では文の感情極性分析を行った。そのため、文の感情極性分析の精度を評価する必要がある。本実験では、文ベクトルを用いた MLRA モデルの評価を行った。本実験では、学習に用いる文の数、文ベクトルの推定に用いる計算方法、および L1 正則化の罰則項のパラメータを考慮した精度を評価した。また、実験におけるコーパスとして、3.2 で説明した楽天公開データを用いた。本研究では、文ベクトルの生成は、TSUKUBA コーパス 4,309 文に加えて、TSUKUBA コーパス以外のデータよりランダムに選択した文を用いた。実験に用いる文の数の合計は、2,060,255 文もしくは 1,265,255 文とした。また、文ベクトルの生成には、階層ソフトマックス (HS) およびネガティブサンプリング (NS) を用いた。実際には、1,265,255 文を HS を用いて学習する実験 (実験 A)、同じ文を NG を用いる実験 (実験 B)、2,060,255 文を HS を用いる実験 (実験 C) を行った。MLRA では、TSUKUBA コーパスを用いて、ブートストラップ法により訓練データとテストデータを 7:1 に分けて評価を行った。罰則項として、L1 正則化を用いた。

感情極性分析の結果を図 3 に示す。図 3 の縦軸は MLRA の精度を示し、横軸は、L1 正則化の罰則項のパラメータの値を示す。また、ラベル A, B, C の付与されたグラフは、それぞれ実験 A, B, C の結果である。図 3 に示すように、HS および NS による計算いずれにおいても、大きな精度の変化はなかった。実際に、いずれの実験においても、最大の精度と最小の精度の差が 2% 程度しか変化しなかった。また、同じ 1,265,255 文を用いた HS による計算と NS による計算では、HS の方が精度が良いため、HS を用いた方が有利だと考えられる。また、

学習に用いる文の数の差分として、5% 近くの精度の差があったため、文の数をさらに増やすことで精度が上がる可能性がある。

また、評価実験で示されたこと以外にも、メモリ使用量を考慮すべきであることがわかった。メモリ使用量に関して、HS の方が NS よりもメモリ領域を使わないことがわかった。実際に、評価実験の際、2,060,255 文の学習を NS を用いて学習を行おうとしたが、空きメモリ領域が十分に確保できなかった。このことから、メモリ使用量においても、NS よりも、HS を用いた方が有利だと考えられる。したがって、この例と評価実験の結果を考慮すると、HS による計算を用いた方がよいと考えられる。このため、本研究では、HS による計算を用いた PV-DM および PV-DBOW のモデルの学習を行うことで、本システムを実装した。

今後の課題として、本研究で生成するスニペットが、検索エンジンに評判情報の閲覧を支援する機能として有用であるかを検討する必要がある。評価実験の方法として、数名の被験者より、通常の検索結果および、感情極性を考慮したスニペットを適用した検索結果が、有用であるかを評価させ、どちらが有用であるかを比較するという方法を検討する。具体的には、いくつかの検索質問に対する、通常の検索結果と、感情極性を考慮したスニペットを適用した検索結果を用意する。被験者は、用意した検索結果のそれぞれが有用か否かを回答する。回答の後には、通常の検索結果が有用という回答数と、感情極性を考慮したスニペットが有用という回答数を基に考察を行う。

## 7. おわりに

本研究では、感情極性を考慮したスニペットを生成し、検索エンジンの検索結果に提示するシステムを開発した。評判情報は見たい情報のみならず、見るべき情報を含む傾向がある。そのため、本研究では、検索エンジンに評判情報の閲覧を支援する機能を付加することを検討した。また、本研究では、文の感情極性を分析する必要があるため、Paragraph Vector を用いた。今後の課題として、感情極性を考慮したスニペットがユーザにとって有用であるか評価する必要がある。

## 参考文献

- [1] Le, Q. V., & Mikolov, T. (2014). Distributed Representations of Sentences and Documents. In Proceedings of The 31st International Conference on Machine Learning, pp. 1188-1196.
- [2] O'Connor, B., Balasubramanian, R., Routledge, B. R., & Smith, N. A. (2010). From tweets to polls: Linking text sentiment to public opinion time series. In Proceedings of the international AAAI Conference on Weblogs and Social Media, pp. 122-129.
- [3] Wang, X., Wei, F., Liu, X., Zhou, M., & Zhang, M. (2011). Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach. In Proceedings of the 20th ACM International Conference on Information and Knowledge Management, pp. 1031-1040.
- [4] Kucuktunc, O., Cambazoglu, B. B., Weber, I., & Ferhatosmanoglu, H. (2012). A large-scale sentiment analysis for Yahoo! answers. In Proceedings of the Fifth ACM International Conference on Web Search and Data Mining, pp. 633-642.