青山学院大学理工学部
情報テクノロジー学科
２０１８年度卒業研究論文

単語分散表現を用いた
コーパス特有の極性辞書構築手法

２０１９年１月２８日提出

指導教員　大原　剛三

１５８１５０７２　橋詰　青弥

目 次

第1章　はじめに

近年，ソーシャルネットワーキングサービス（SNS: Social Networking service）や Web サイトの持つ情報が増え，それに伴って自然言語処理やウェブマイニングにおける感情的分析の需要が高まっている．感情的分析とは，ある対象について書かれた文章から，その文章を書いた人間の感情や対象への評価を推測することである．レビューサイトやブログなど，日常的に接する文章情報が増えている現在，我々の身近にあるその膨大な情報を解析することは，研究目的だけでなく商業的にも大きな価値があると言える．具体的には，商品のリコメンドやある利用者の生活習慣や利用方法の解析など，応用できる範囲は非常に広いと言えるだろう．通信販売サイトの商品についての消費者レビューについて，肯定的か否定的かの統計情報をとるだけでも利用者にとってよい情報になることは想像に難くない．このように，多くの事柄がインターネット上で文章やメールでやり取りされている現在，文章の感情解析はとても大きな課題となっている．しかし，利用者の語彙，つまり解析対象となる単語数は増える一方であり，毎年のように新語が生まれている．また，既存の単語についても使い方や感情的な意味が変わることさえある．単語の意味や使い方を現状に即した形で利用する技術の必要性は高まり続けているが，その意味や使い方を常に反映させた辞書を維持することは，常に更新され続ける世の中の文章を解析，評価することになり膨大な労力を要する．そこで本研究では，対象とする文章コーパスに依存した形で単語レベルの感情的な意味を自動的に抽出することを目的とする．

文章に対する感情分析では，文章中に表れる単語やその関係に基づき文全体に対する感情を判定する．そのために単語それぞれが持つ感情的な極性（ポジティブ/ネガティブ）を考える必要があり，その極性を登録した極性辞書が必要となる．たとえば，一般的に “good” という単語は良い意味を持つ文章に用いられやすい．そのような情報を統計的にまとめ，極性辞書として利用する．極性辞書は文章に出現する極性に関わる語彙をなるべく多く包含し，解析対象の文章に適用できるよう更新されている必要がある [1]．WordNet や，シソーラスを繋いで作成した言語ネットワークなどの既存の言語資源を基に辞書を構築するモデル [2, 3] やリンクデータを基にして感情極性を予想するモデル [4]，初めから一定数の極性がわかっている単語群を小さな辞書とし，それを拡張して極性辞書を構築するモデル [5, 6] が提案されている．感情極性ではなく，評価文に対する評価表現の獲得としては，HTML（HyperText Markup Language）文書から評価表現辞書を自動構築する研究がある [7]．また，否定文のように文脈によって文全体の極性は変わる．そのため，単語レベルの極性判断だけではなく，単語の極性が文脈によって反転することを考慮したモデルも提案されている [8]．その他，単語分散表現を用いた文章の極性評価に関する研究としては，評価極性をオートエンコーダーによって分類した研究 [9]，感情極性を単語分散表現中に埋め込んだ研究 [10] などが存在している．これらの研究では，極性反転の起こるパターンを考

慮して既存の小さな辞書を拡張していく手法が多い．つまり初期情報としていくつか人手で定められた極性を基にしている以上，客観的な評価は難しく，完全にコーパスに依存した表現や感情極性に対応できているとは言い難い．初期情報の極性の正確性が解析対象のコーパスについても必ず正しいとは断言できないからである．

そこで，本研究ではコーパス特有の感情極性辞書の構築を提案する．すなわち，事前に極性の定められている単語はないことを仮定し，ラベル付きコーパスのみを用いて感情極性辞書を構築することを試みる．ラベル付きのコーパスからある単語に極性情報を付与するためには，その単語周辺の単語情報，すなわち単語の出現した文脈が極めて重要である．文脈に応じて使われ方が変わる単語は，その文の極性ラベルに応じて周辺単語が大きく変わると考えられる．本研究は，周辺単語から，学習元コーパスによる使われ方の違いを検知することができることに着目した．その周辺単語情報を分散表現を用いて処理し，極性を付与しやすいと考えられる単語を選定，極性を付与する．そして分散表現を用いてその他の単語の極性を予測する形で辞書を構築する手法を提案する．

以下，2 章では感情極性辞書の関連研究，及び本論文における主要な役割を持つ単語分散表現の関連研究を説明する．3 章では提案手法を説明し，4 章にて評価実験の結果・考察を述べる．5 章で本論文をまとめる．

第2章　関連研究

2.1　緒言

単語を極性を考慮した表現，すなわち感情極性に変換し，文章分類や感情推定などのタスクに応用する研究は多く存在する．その感情極性の付与に関しては，人手で極性を判定し単語に付与する研究をはじめ，多くの研究がなされてきた．また，単語感情極性を利用した多くの研究の目的は文章分類である．近年は，単語感情極性の推定や文章分類タスクに単語分散表現を用いた手法が注目され，その性能の向上に大きく貢献している．以下では，感情極性辞書について 2.2 節で説明する．2.3 節では，本研究で用いた単語分散表現についての概要を説明する．単語分散表現を用いた文書分類タスクについての関連研究については 2.4 節で述べる．

2.2　感情極性辞書についての関連研究

単語それぞれを数値として表現する研究は従来から存在する．その中でも，文章の感情を推定，分類するために付与した数値と単語のセットをまとめたものは感情極性辞書と呼ばれている．文章分類タスクに応用する際には，分類したい文章中に出現する語彙の多くが辞書に登録されている必要があり，辞書は大規模であることが求められる．また，文脈によっては同一単語でも感情的な意味合いが大きく変わることがあることが以前から課題として存在している．大規模で実用的な感情極性辞書構築のための研究としては，Kamps らの WordNet を用いたシソーラスによるネットワークを利用したもの [3]，Turney らの検索エンジンを用いた単語間の共起関係と検索ヒット数による感情極性辞書の拡張 [11] が挙げられる．これらの研究は，一定数の極性が判明している単語があれば自動的に感情極性辞書を構築することを可能にした．立石らは，Web から評判情報を抽出するため，ドメインごとに商品カテゴリ（書籍・コンピュータなど）と評価表現（好き，不安定など）を集めた評価辞書を人手で作成した [12]．また小林らは，対象・属性・評価値の組を評価表現としてそれぞれを抽出し，人手でその組が評価表現として適切か否かを判断，評価値を付与することで辞書を作成した [13]．これらは，評価表現を抽出することで文脈による極性の変動に対処しているが，人手による作業が極めて多く，大規模な辞書をドメインごとに構築することに大きなコストを伴うという課題が残る．

池田らは，文脈における極性的な意味合いの変化という課題に対し，単語極性の反転を検知し評価するモデルを提案している [14]．文中に出現した単語とその文の極性が一致しないのであれば，その単語は極性が反転していると考え，その反転を 2 値分類タスクとして予測し，感情極性辞書を用いた文章分類の精度を向上させた．中村ら [10] は，既存の感情極性辞書を基に単語分散

表現を初期化した上で分散表現を再度学習し，感情極性を考慮した分散表現を獲得する手法を提案している．これらの研究により，文脈による極性反転を考慮することが可能になり，また，既存の辞書を基に拡張・改良する手法が提案されたため，感情極性辞書を用いた文章分類タスクの精度は向上した．

2.3 単語分散表現についての関連研究

単語をベクトルに置き換え単語同士の関係を表現する手法は，自然言語処理の分野における多くのタスクに大きく貢献している．たとえば，佐藤ら [5] は，コーパスから獲得した単語分散表現を単語の素性として，SVM（サポートベクターマシン）に学習させることにより，少量の極性情報が判明している単語を基に大規模な感情極性辞書拡張を実現した．このような単語をベクトルに置き換える手法は，周辺単語の情報を考慮する際に有用であり，本節では，その手法について説明する．

最も簡単なベクトル化の手法と言えるものは One-hot ベクトルによる表現である．語彙の全単語に固有な次元を割り当て，その単語に対応する次元の要素を 1，それ以外を 0 として表現する．語彙数が 5 あれば 5 次元のベクトル表現になり，2 番目の単語のベクトルは (0, 1, 0, 0, 0) と表現される．一方，文脈の情報を用いて文や単語の生成確率を予測し，単語をベクトル化するモデルを言語モデルと呼ぶ．そして，入力と出力を one-hot ベクトルとして，特に単語の生成確率をニューラルネットワークによって予測するモデルをニューラル言語モデルと呼ぶ．一般に，ニューラル言語モデルは計算コストが高いことが課題として存在し，Mikolv らによる計算コストを抑えたモデル [15] が Word2vec として現在広く用いられている．Word2vec には CBoW（Continuous Bag-of-words）と skip-gram の 2 つのモデルが存在する．それぞれの出力には one-hot ベクトルが用いられ，CBoW は入力として周辺単語の one-hot ベクトルのリスト，skip-gram は入力として単一単語の one-hot ベクトルが図 2-1 に示すようにそれぞれ用いられる．コーパス中の語彙全体の集合を V として，このコーパスに対して単語分散表現を獲得すると仮定する．語彙の t 番目の入力単語を表す one-hot 列ベクトルを $\boldsymbol{x}_t$，その入力単語 $\boldsymbol{x}_t$ に対応して実際に獲得される単語埋め込みベクトル（列ベクトル）を $\boldsymbol{e}_t$ とする．また，同様に語彙の r 番目の出力単語を表す one-hot 列ベクトルを $\boldsymbol{y}_r$ と定義し，$\boldsymbol{y}_r$ に対応する単語埋め込みベクトル（列ベクトル）を $\boldsymbol{o}_r$ とする．これらの単語埋め込みベクトルのベクトルサイズを s とすると，$\boldsymbol{e}_t, \boldsymbol{o}_r$ は語彙 V の数だけ存在し，$|V|$ 行 s 列で表現される行列 E，O を用いて式 (2.1) 及び式 (2.2) で定義される．

$$E\boldsymbol{x}_t = \boldsymbol{e}_t \tag{2.1}$$

$$O\boldsymbol{y}_r = \boldsymbol{o}_r \tag{2.2}$$

CBoW を用いたモデルにおいて，入力は周辺単語の one-hot ベクトルのリストであり，それを H と定義する．出力語彙 V 中の t 番目の単語 $\boldsymbol{y}_t$ が出力される確率 $P(\boldsymbol{y}_t|H)$ は式 (2.3) 及び式

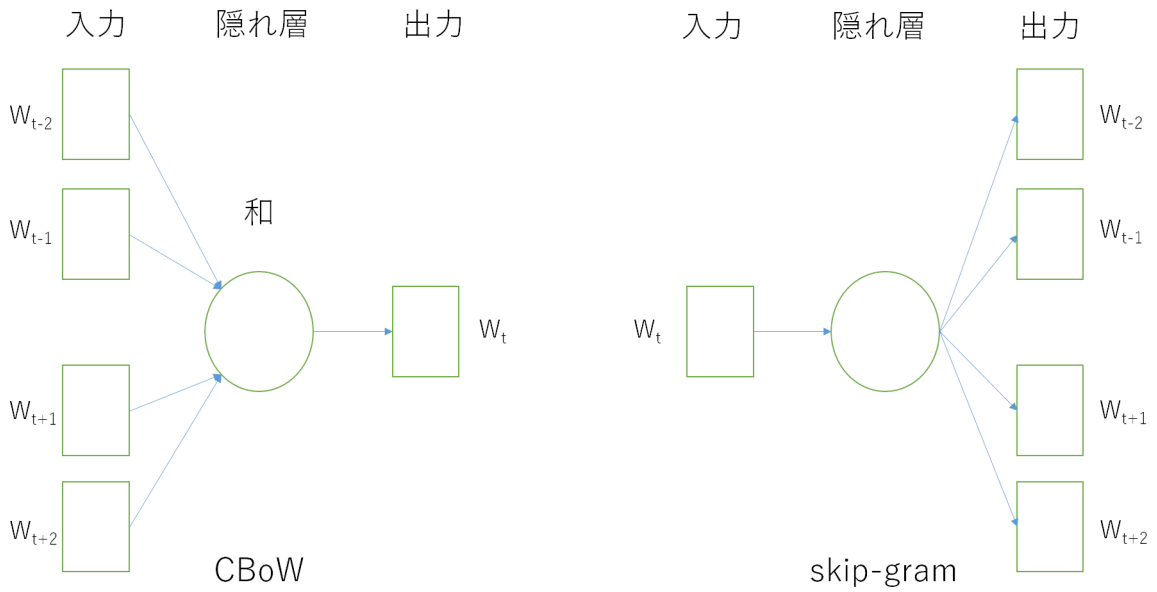


図 2-1: CBoW 及び skip-gram の入出力

(2.4) で表せる．

$$P(\boldsymbol{y}_t|H) = \frac{\exp(\phi(H, \boldsymbol{y}_t))}{\sum_{\boldsymbol{y}'_t \in V} \exp(\phi(H, \boldsymbol{y}'_t))} \tag{2.3}$$

$$\phi(H, \boldsymbol{y}_t) = \sum_{i:\boldsymbol{x}_i \in H} (E\boldsymbol{x}_i) \cdot (O\boldsymbol{y}_t) \tag{2.4}$$

skip-gram を用いたモデルにおいて，単語 $\boldsymbol{x}_t$ を入力としたとき，その単語の周辺単語 $\boldsymbol{y}_r$ が出力される確率 $P(\boldsymbol{y}_r|\boldsymbol{x}_t)$ は式 (2.5) と式 (2.6) で表される．

$$P(\boldsymbol{y}_r|\boldsymbol{x}_t) = \frac{\exp(\phi(\boldsymbol{x}_t, \boldsymbol{y}_r))}{\sum_{\boldsymbol{y}'_r \in V} \exp(\phi(\boldsymbol{x}_t, \boldsymbol{y}'_r))} \tag{2.5}$$

$$\phi(\boldsymbol{x}_t, \boldsymbol{y}_r) = (rE\boldsymbol{x}_t) \cdot (O\boldsymbol{y}_r) \tag{2.6}$$

CBoW を用いたモデルにおいて，入力は周辺単語の one-hot ベクトルのリスト H であり，そのリストを作成する際に窓の大きさ，すなわち前後にある単語のうちいくつを周辺単語のリストとして抽出するかを考慮する必要がある．もし $H = \boldsymbol{x}_t$ とすれば，式 (2.3) は式 (2.5) と一致する．つまり，CBoW による式 (2.3) は skip-gram による式 (2.5) を包含していると考えることができる．

次に，学習時の目的関数について説明する．CBoW による学習では，式 (2.3) を最大化することを考える．そのため，式 (2.3) の対数尤度をとり，-1 をかけ，最小化問題として設定した式 (2.7) を目的関数として使用する．skip-gram についても同様に式 (2.5) から導出した式 (2.8) を目的関数とする．

$$L(E, O|H) = -\sum_{(H, \boldsymbol{y}_t) \in D} \log(P(\boldsymbol{y}_t|H)) \tag{2.7}$$

$$L(E,O|H) = -\sum_{(H,\boldsymbol{y}_t)\in D} \log(P(\boldsymbol{y}_t|\boldsymbol{x}_t)) \tag{2.8}$$

しかし，内積及び指数関数の計算が $|V|$ 回求められる式 (2.7) を目的関数に設定すると計算量が膨大になり，大規模な学習データに対応できない．そのため，計算量を削減するための手法として，負例サンプリング（negative sampling）と階層的ソフトマックス（hierarchical softmax）が提案されている．負例サンプリングでは，訓練データのほかにノイズとして別の確率分布から生成されたデータを考える．学習データの一つ $(H,\boldsymbol{y}_i)$ が実際の訓練データを生成する確率分布 P^D から生成される確率 $P((H,\boldsymbol{y}_t)\sim P^D)$ はシグモイド関数を用いて式 (2.9) のように表せる．

$$P((H,\boldsymbol{y}_t)\sim P^D) = \frac{1}{1+\exp(\phi(H,\boldsymbol{y}_t))} \tag{2.9}$$

同様に，ノイズデータを生成する確率分布 $P^{D'}$ から $(H,\boldsymbol{y}_t)$ を生成するモデルを $P((H,\boldsymbol{y}_t)\sim P^{D'})$ とすると，実データを生成する確率分布 D から生成されたデータなのか，ノイズデータを生成する確率分布 D' から生成されたデータなのかの識別問題の目的関数は式 (2.10) と表される．

$$L(E,O|D,D') = -\sum_{(H,\boldsymbol{y}_t)\in D} \log(P((H,\boldsymbol{y}_t)\sim P^D)) + \sum_{(H,\boldsymbol{y}_t)\in D'} \log(P((H,\boldsymbol{y}_t)\sim P^{D'})) \tag{2.10}$$

一方，階層的ソフトマックスとは $|V|$ 個あるラベルを階層的に分類する，すなわち二値分類を連続的に行うことで階層的に分類する手法である．全語彙の中から選択するのではなく，ラベル数を限定し，その中で分類することを繰り返すことにより，計算量を削減する．葉ノードが $|V|$ 個あり，それぞれの葉が単語に一意に対応している 2 分木を考えた場合，葉ノード以外のノードは $|V|-1$ 個存在する．ある語彙 y を選択したときの経路長を $L(y)$ とし，経路上の各ノードのインデックスを $(\pi_1(y), ..., \pi_{L_{(y)}}(y))$ とする．ルートノードから葉ノード y までの経路において，子ノードのどちらを辿ってきたかを $\{0, 1\}$ で記録し，それによって得られたビット列を $(\boldsymbol{b}_1(y), ..., \boldsymbol{b}_{L(y)}(y))$ と定義する．各単語が確率的に選択されるとすると，ルートノードから単語 y を選択する確率は各符号を選択する確率の積として式 (2.11) のように表される．

$$P(y) = \prod_{j=1}^{L(y)} p(\pi_j(y), \boldsymbol{b}_j(y)) \tag{2.11}$$

さらに，ロジスティック回帰モデルを $p(\pi_j(y), \boldsymbol{b}_j(y))$ に適用し，各ノードに対応する特徴ベクトルを $f_\theta(\pi_j(y))$ としたとき，ビット列を {0,1} ではなく {-1,1} によって書き直すことで，式 (2.11) は式 (2.12) によって近似することができる．

$$p(\pi_j(y), \boldsymbol{b}_j(y)) = \text{sigmoid}((2\boldsymbol{b}_j(y)-1)f_\theta(\pi_j(y))) \tag{2.12}$$

$$= \frac{1}{1+\exp((2\boldsymbol{b}_j(y)-1))f_\theta(\pi_j(y))} \tag{2.13}$$

階層的ソフトマックスにおいては，$P(y)$ の負の対数尤度を最大化するように最適化を行うため，2 分木の経路長分の計算量になる．すなわち $|V|$ の対数程度の計算量で抑えることが可能な

手法である．skip-gram において，入力単語 $\boldsymbol{w}_i$ の持つベクトル $\boldsymbol{e}_i$ の学習は，入力文を ($\boldsymbol{w}_{i-2}$，$\boldsymbol{w}_{i-1}$，$\boldsymbol{w}_i$，$\boldsymbol{w}_{i+1}$，$\boldsymbol{w}_{i+2}$) としたとき，$\boldsymbol{w}_i$ 以外の単語 ($\boldsymbol{w}_{i-2}$，$\boldsymbol{w}_{i-1}$，$\boldsymbol{w}_{i+1}$，$\boldsymbol{w}_{i+2}$) の持つベクトル ($\boldsymbol{o}_{i-2}$，$\boldsymbol{o}_{i-1}$，$\boldsymbol{o}_{i+1}$，$\boldsymbol{o}_{i+2}$) 同士の演算結果を単語の意味的距離として考える．意味的距離を誤差として逆伝播させることで学習を行っている．

2.4 単語分散表現を用いた文章分類

単語分散表現は多くのタスクに応用され，文章分類では RNN（Recurrent Neural Network）を用いた研究 [16] や Kim による畳み込み層を持つ DNN（Deep Neural Network）モデル [17] を用いたものが挙げられる．畳み込み層を持つ DNN は Alex らによる画像の分類モデル [18] があり，画像分類のための手法として有名である．畳み込みとはデータを局所的に演算し特徴とするものであり，チャネルが 1 である画像を例にすると式 (2.14) で表される．

$$Y_{ij} = \sum_{s=0}^{m-1} \sum_{t=0}^{n-1} W_{st} X_{(i+s)(j+t)} + b \tag{2.14}$$

ただし，入力として行列 X を仮定し，行列 W は重み，b はバイアスである．m，n は畳み込む際に注目する局所領域の大きさによって決定される定数であり，i，j は出力の大きさによって決定される．これにより，行列から特徴として 1 つの値を獲得することが可能になる．Kim による研究 [17] は，単語分散表現を文ごとに並べたものを入力として扱い，畳み込みによる特徴抽出，そして文章分類の精度向上に貢献した．

2.5 結言

本章では，感情極性辞書を構築するにあたっての関連研究，及び本研究で使用する単語分散表現についての関連研究を説明した．感情極性辞書を利用するにあたっての課題としては，大規模であること及び文脈を考慮しなくてはならない点が挙げられる．既存の研究においてこれらの課題に対処する際，外部の言語資源に頼っていることや人手による対処に伴う大きなコストが課題となっていた．外部の言語資源に頼る場合には辞書が解析したい文章特有の表現や使いまわしに対応できないという課題が残る．本論文ではそれらの課題を解決するため，ラベル付きコーパスから外部の言語資源に頼らず感情極性辞書を獲得することを試みる．次章では提案手法の詳細について説明する．

第3章　提案手法

3.1　緒言

前章において，感情極性辞書についての関連研究及び単語分散表現について説明した．感情極性辞書のもつ課題として，大規模な構築が困難である点と文脈によって極性が変わるという点があった．既存研究では，それらを外部言語資源に頼った形で解決しており，文章の種類や抽出元に特異な表現，言い回しを考慮していない．本研究では単語分散表現を用いてラベル付きの文章から単語それぞれの感情極性を考慮した分散表現を獲得し，佐藤らの手法 [5] に従い，感情極性辞書を構築する．佐藤らの手法は，極性が判明している一部の単語を設定し，そこから単語極性辞書を拡張・構築する手法であるが，本研究では，極性が判明している単語がなくても辞書を拡張・構築が可能な手法を提案する．本論文の核をなすものは，文脈によって極性の変化する単語の選定と，それらの単語を文脈によって分割し，それらの単語の極性を定義することである．以下では，3.2 節で提案手法の概要を述べ，3.3 節で分割する単語の選定基準，3.4 節では単語の分割・置換方法を説明する．3.5 節ではそのようにして獲得した単語置換後のコーパスについて述べ，3.6 節にて感情極性辞書の構築手法を述べる．3.7 節にて結言を述べる．

3.2　提案手法の概要

提案手法全体の流れを図 3-1 に示す．本手法は，1）分散表現の獲得，2）分割する単語の選定，3）単語の分割・置換，4）単語極性の予測の 4 段階に大別できる．提案する手法の基本的なアイデアは，ラベル付き文章から単語を形式上二つに分割し，その上で分散表現を獲得することにある．ポジティブ，ネガティブのラベルが付与されている文章内には同一の単語であってもポジティブな使い方，ネガティブな使い方の両方で使われているものがある．すなわち，すべての単語は極性的な多義性を持つ可能性があると考えられる．文脈によって極性の変化が発生する単語を，分散表現から読み取れる周辺単語に基づいて検知し，各極性に対する単語へ形式上分割する．元のコーパス上にあるその単語を対応する分割後の単語で置換した上で，コーパス全体から獲得した分散表現は，極性情報が考慮された表現と考えられる．分割した単語は，コーパスのラベルに対応する極性を保持していると考えられ，それらの単語を基にして感情極性辞書を拡張・構築することが可能である．分割した単語を教師として，佐藤ら [5] の手法に従い感情極性辞書を構築する．

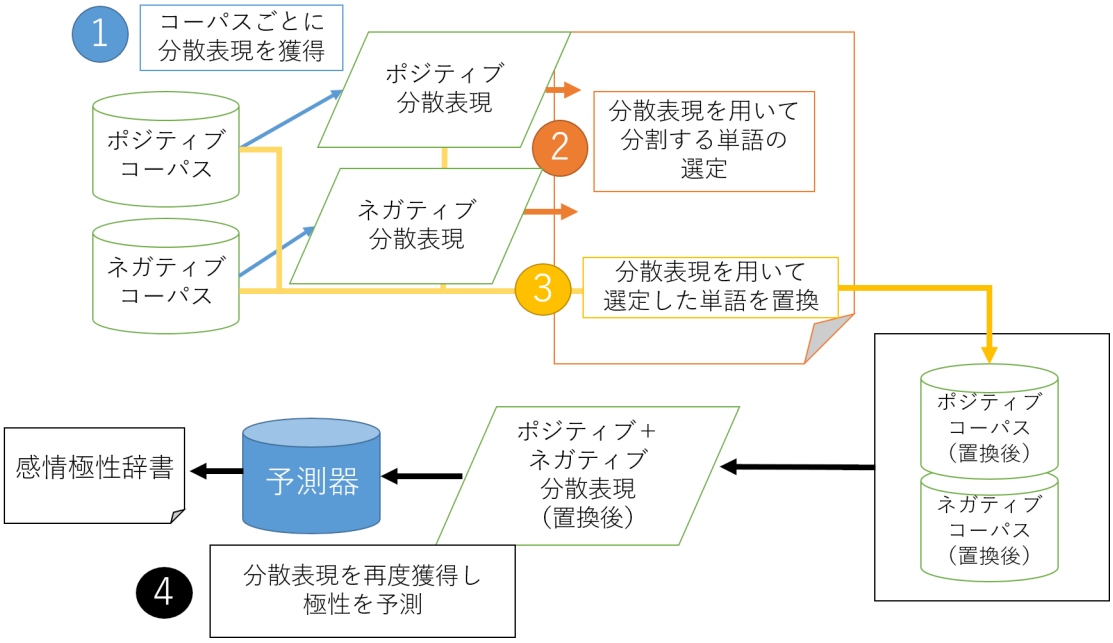


図 3-1: 提案手法の概要

3.3 分割する単語の選定

文中に出現する単語の中で，文脈によって，極性，または使われ方が変化しやすいと考えられるものを選定する．常に極性的に一つの意味でのみ使われやすい単語は分割せず，文脈によって使われ方が大きく変わる単語のみを選定することによって，局所的な使われ方による汎用性の低い分割を避けることができると考えられる．初めに，ポジティブコーパス，ネガティブコーパスそれぞれの分散表現を獲得し，それぞれを $\boldsymbol{r}_p, \boldsymbol{r}_n$ とする．なお，以下では特に断りがない限り，分散表現を獲得する際は Word2vec の CBoW のモデルを用いており，計算量の削減のため階層的ソフトマックスを利用するものとする．次に，文脈によって極性が大きく変わる単語を分割する．文脈に応じて使われ方が変わる単語は，その文のラベルに応じて周辺単語が大きく変わると考えられる．すなわち，似た使い方をする単語の類似度が高くなるように学習された分散表現であれば，周辺単語に着目することで，学習元コーパスによる使われ方の違いを検知することができる．本研究では，ある単語の極性が反転しやすいか否かの指標として，それぞれの分散表現でコサイン類似度 C の高い単語を複数列挙した際に，重複しなかった単語の個数を用いた．ここで，$\boldsymbol{v}$，$\boldsymbol{w}$ をコサイン類似度を算出したい単語に対応する分散表現としたとき，$\boldsymbol{v}$，$\boldsymbol{w}$ のコサイン類似度 $C_{\boldsymbol{v},\boldsymbol{w}}$ は式 (3.1) によって定義される．

$$C_{\boldsymbol{v},\boldsymbol{w}} = \frac{\sum_{i=1}^{|\boldsymbol{v}|} v_i w_i}{\sqrt{\sum_{i=1}^{|\boldsymbol{v}|} {v_i}^2} \cdot \sqrt{\sum_{i=1}^{|\boldsymbol{v}|} {w_i}^2}} \tag{3.1}$$

たとえば，“good” という単語に，ポジティブコーパスから獲得した分散表現 $\boldsymbol{r}_p$ においてコサイン類似度が高い単語を列挙すると，“up”，“charming”，“scene” となり，同様にネガティブコーパスから獲得した分散表現 $\boldsymbol{r}_n$ においては “watch”，“scene”，“waste” となると仮定する．こ

のとき，“scene” のみ共通してコサイン類似度が高いが，それ以外は重複していない．Word2vec は似通った使われ方をしている単語のコサイン類似度が高くなるように学習した分散表現を獲得するため，ここで重複している単語の個数が多いほど，両コーパスでのその単語の使われ方が似通っていると考えられる．語彙中のすべての単語について，いくつ重複するかを算出し，重複する単語の少ない順に優先度を設定することで分割する単語の選定を行う．

3.4 単語の分割

分割する単語について，その単語がポジティブ，ネガティブどちらの使われ方をしているかを，その単語の出現した文から判断する必要がある．その判断のための指標の算出方法を 2 つ提案する．

3.4.1 ベクトル積による判断

まず，式 (2.3) を考えると，獲得した分散表現は，周辺単語とのベクトル積を正規化して得られる生起確率を最大化した結果得られるベクトルと解釈できる．つまり，ある単語が出現した文中の，前後数単語とのベクトル積を考えることにより，その分散表現における，その単語の生起確率に相関のある値を獲得することができる．分割する単語の分散表現を $\boldsymbol{v}_t$，その単語が出現した文中の前後の単語の分散表現のリストを H とすると，分割する際の指標 P_t は式 (3.2) により定義できる．

$$P_t = \sum_{i:\boldsymbol{v}_i \in H} (\boldsymbol{v}_i) \cdot (\boldsymbol{v}_t) \tag{3.2}$$

この指標 P_t をポジティブ分散表現及びネガティブ分散表現の両方について算出し，比較した際に大きかった方の使われ方であると判断する．分割する際には単語の末尾に “-p” もしくは “-n” などの識別子を付与し，以後，別の単語として扱う．そのようにしてポジティブコーパス及びネガティブコーパス内の分割可能な単語すべてを分割し，置換したコーパスをポジティブコーパス（置換後）及びネガティブコーパス（置換後）とする．

3.4.2 ニューラルネットワークによる判断

分散表現を獲得する際には，単語の生起確率を最大化する．そのため，分散表現を獲得した際のニューラルネットワークを用いて周辺単語から生起される可能性の高い単語を導出することができる．ある単語が出現した文中の前後数単語を入力として，ポジティブ，ネガティブ両方の分散表現を獲得した際のニューラルネットワークにそれぞれ入力する．そのとき出力される生起される確率の高い単語群を比較することで，その文中における単語の使われ方を判断できる．本手法では，それぞれの分散表現において，単語の生起確率が全語彙中下から何番目であったかを指標として定義する．同様にして，この指標を基にコーパス内の単語を置換する．

3.5 置換後のコーパス

前節において単語を置換した後のコーパス全体から，単語の分散表現を獲得する．この際，ポジティブコーパス（置換後）及びネガティブコーパス（置換後）両方を同時に訓練データとして用いる．単純にコーパス全体を訓練データとして獲得した単語分散表現と比較すると，置換後のコーパスから獲得した分散表現は，文脈によって極性が決定される単語を考慮した語彙を持つと考えられる．理想的な分割・置換がなされていれば，文脈によって使われ方の変わる単語はすべて分割され，文脈による単語の使われ方の変化がより少なくなると考えられる．たとえば，前節で分割した際にポジティブな使い方として “masterpiece-p”，ネガティブな使い方として “masterpiece-n” と置換された単語を考える．“masterpiece-p” はポジティブな使い方のみを仮定され分割しているため，この単語の極性はポジティブである可能性が高いと言える．すなわち，ラベル付きコーパスから，一定数の極性が付与された単語を獲得したと考えられる．

3.6 感情極性辞書の構築

佐藤ら [5] の手法は，既存の極性辞書に含まれる単語を訓練データとして，単語分散表現を素性に教師あり 2 値分類を行うものである．本研究においては，佐藤らの手法のうち，分散表現の獲得方法及び訓練データを変更し，SVM による教師あり 2 値分類を行う．単語分散表現を素性とする点は変わらないが，既存の極性辞書に代わり，前節で置換した単語を教師データとして学習する．次に，学習したモデルを用いて，置換していない単語についても極性を予測する．置換した単語は前節で説明した通り，ラベルの各極性に対応していると考えられ，それを教師として学習することによって，コーパス特有の感情極性辞書を構築できると考えられる．

3.7 結言

本章では，提案手法の概要及び中心的な処理について述べた．分割する単語の選定及びその分割基準を適切に設定することで，ラベル付きコーパスから単語の極性情報を部分的に抽出することが可能になる．その単語を教師とすることで，コーパス特有な感情極性辞書を構築することが可能である．次章では，提案手法における分割の妥当性を示すための実験，及び既存手法を用いた比較実験の手順，結果について述べ，考察を行う．

第4章　評価実験

4.1　緒言

前章では，提案手法の詳細について述べた．提案手法ではポジティブ，ネガティブそれぞれのコーパスから獲得した分散表現を用いて，分割する単語の選定，単語の分割・置換を行う．そして置換した単語を教師として，感情極性辞書を構築する．本章においては，提案手法によって獲得した単語分散表現について，公開されているデータセットを用いた比較実験を行い，単語の分割手法の検証を行う．以下，4.2 節では分析対象のデータセットについて述べ，4.3 節では本検証実験の方法，4.4 節にて実験結果及び考察を述べる．

4.2　分析対象データセット

本実験ではテキストデータセットとして Movie Review Data コーパス[1]を用いる．Movie Review Data コーパスとは，映画の評価文をポジティブな感想かネガティブな感想かに分類し，クリーニング処理を行った英文テキストデータコーパスである．ポジティブ・ネガティブな感想としてそれぞれ 5,331 文が収録され，合わせて 10,662 文ある．本実験においては単語の極性に関する検証のみを目的とするため，極性情報が一般に付与されにくいと思われる品詞を削除し，名詞・形容詞・動詞・副詞の 4 品詞のみに限定した．また，単語を基本形に直すために，Python のモジュールである nltk[2]から wordnet[3]の情報を用いて基本形に直す WordNetLemmatizer を用いた．名詞を複数形から単数形に統一する基本形への変換を行い，単語の感情極性を考える上で処理しやすいデータに加工した．

4.3　実験方法

4.3.1　評価実験の流れ

評価実験全体の流れを図 4-1 に示す．単語の分割時の基準が妥当であったかを評価するために，まず分割・置換後のコーパスから評価用分散表現（置換後）を獲得する．評価用分散表現（置換後）を，置換前のコーパスから獲得した評価用分散表現（置換前）と比較することにより，極性情報がよりよい形で埋め込まれているかを評価する．提案手法における単語の置換が適切である

[1] https://www.cs.cornell.edu/people/pabo/movie-review-data
[2] http://www.nltk.org
[3] https://wordnet.princeton.edu

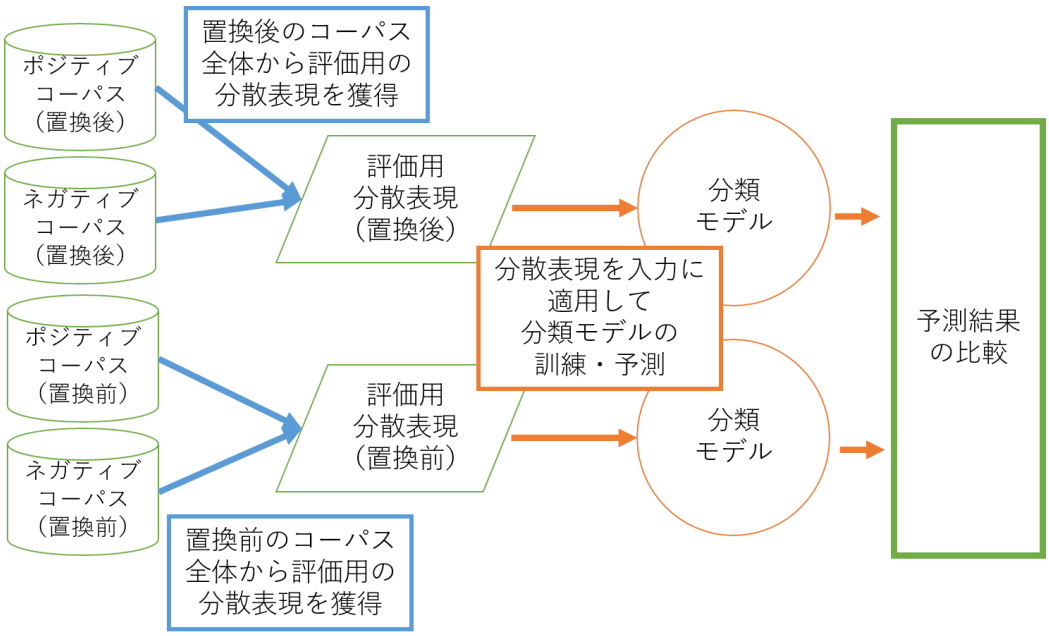


図 4-1: 評価方法

と仮定する．このとき，置換された語彙は，文脈によって使われ方の変化が発生しにくい語彙になっていると考えられる．極性反転は文章分類の際の課題であり，精度を落とす原因であった．そのため，提案手法が適切に単語を置換しているのであれば，提案手法によって獲得された分散表現による文章の極性分類は，置換前のコーパスから獲得した分散表現による極性分類よりも精度が高いと考えられる．そこで，評価用分散表現（置換前）と，評価用分散表現（置換後）を用いて文章分類を実施する．分類モデルとしては，Kim による畳み込み層を持つモデル [17] を利用する．入力として，入力文の単語に対応する分散表現を行列として用いる．すなわち，入力文の単語数を n，分散表現のベクトルサイズを s とすると，入力は n 行 s 列の行列である．出力は，ポジティブな文章かネガティブな文章かを $\{0, 1\}$ によって示した値である．また，検証は 10 分割交差検証を用いて行った．これはまず，置換前のコーパスを 10 個に等分割し，1 つをテストデータ，残りを訓練データとして評価実験を行う．これを 10 回繰り返し，毎回テストデータとして扱われる分割後のデータを変更する評価方法である．そのため，データの一部のみに依存した評価を避けることが可能であり，モデルの妥当性をより正確に検証できる．テストデータのうち，半分は分類モデルの学習時のバリデーションデータ（学習時評価用データ）とし，残り半分を学習後のモデル評価用データとして利用した．

4.3.2 分割する単語の選定手法の評価

単語の分割を行うため，分割する単語の選定を行う．分割する単語選定時の優先度として比較する周辺単語の数は，ポジティブコーパス及びネガティブコーパスに共通する語彙数の 1%とした．また．分割する単語の選定方法の妥当性の検証のための比較手法として，単語の出現回数が多いほど優先して分割されるような基準を設定した．

4.3.3 単語の分割・置換手法の評価

分割・置換する単語数はポジティブ・ネガティブコーパスに共通する語彙数の 1%，5% で実験した．また，単語の分割をベクトル積に基づいて行う際には，式 (3.2) に示すように文中の前後の単語を用いて算出するが，ベクトル積をとる単語は，分割する単語の前後 2 単語のみに限定した．また，出現回数が 10 回に満たない語彙は極性情報を特定できるほどの文脈情報を取得できないとして除外した．さらに，分割した単語で置換する際には，ポジティブな使われ方と判断した単語には “-p”，ネガティブな使われ方と判断されたものについては “-n” を末尾に付与し，以後，別単語として扱った．

4.3.4 分散表現獲得時のパラメータ

本実験において，提案手法と同様の前処理を施したコーパスから獲得した分散表現と，提案手法によって獲得した分散表現が実験対象となる．分散表現を獲得する際のモデルは Word2vec のうち CBoW のモデルを用い，階層的ソフトマックスによる計算量削減を行う．共通するパラメータとして，モデルの参照する文脈の範囲（ウィンドウサイズ）は 5，獲得する分散表現の次元数は $\{100, 300\}$，分散表現の語彙となるための最小出現回数は 1 に設定する．それ以外の分散表現に関するパラメータは Python のモジュールである gensim[4]の初期設定を用いた．また，獲得した分散表現を用いて文の極性評価を行う際のモデルとして Kim によるモデル [17] を利用し，分類モデルについてのパラメータもそれに倣った．

4.4 実験結果及び考察

4.4.1 分割・置換する単語選定に関する結果

分割・置換する単語の選定方法に関する実験結果を表 4-1 に示す．出現頻度の高いものから分割する手法を比較手法とした．また，単語の使われ方の判断方法は 3.4.1 節のベクトル積による判断手法を用いた．獲得する分散表現の次元，ベクトルサイズを {100，300}，分割する単語数をポジティブコーパス及びネガティブコーパスに共通する語彙数の {1%，5%} として検証を行った．スコアとして，10 分割交差検証を行った際の正答率（Accuracy）の平均値を小数第 4 位で四捨五入したものを記載している．ここで，Accuracy は N 件のテストデータに対して，モデルの予測値と真の値が一致した件数を t 件としたとき，式 (4.1) によって定義される．

$$\text{Accuracy} = \frac{t}{N} \tag{4.1}$$

この結果から，単語の選定方法，単語の置換の両面においてベクトルサイズを 300 にした提案手法のスコアが最も高かった．しかし，提案手法において，分割する単語数の上昇とともにスコアが低下することを確認した．この結果から，適切に分割できているものは一部であると考えら

[4] https://github.com/RaRe-Technologies/gensim/blob/develop/gensim/models/word2vec.py

表 4-1: 分割・置換する単語選定に関する評価実験の結果

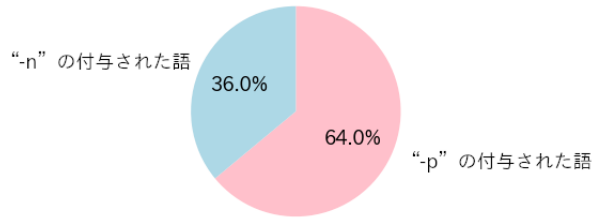
単語選定方法	ベクトルサイズ	単語分割数	スコア
提案手法	100	1%	0.753
提案手法	100	5%	0.742
提案手法	300	1%	**0.763**
提案手法	300	5%	0.737
頻度	100	1%	0.649
頻度	100	5%	0.599
頻度	300	1%	0.641
頻度	300	5%	0.603
分割無し	100	-	0.748
分割無し	300	-	0.748

れる．すべての単語に対して極性の変化を考慮できている分割手法ではなく，前後の文脈が大きく変わる単語のみの単語極性の変化を考慮できたと思われる．否定や皮肉などの言い回し，すなわち単語列としては一部の変化であったり，極性反転の原因が直近の単語ではない単語には提案手法が対応できていないと考えられる．これは単語分散表現が，周辺単語を用いて該当単語の生起確率の最大化をなすことによって獲得されるために，係り受けや単語の順序を考慮できない点が影響したと思われる．すなわち，汎化性能が高い反面，周辺単語の微細な変動による意味の変動を考慮できていない可能性があり，単語の極性的な使われ方を完全に考慮することは難しいと考えられる．

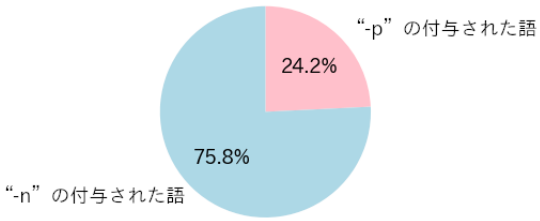
次に，分割する語彙数を全体の 1%，ベクトルサイズを 300 にした時の単語の置換結果をコーパスごとに示す．置換した単語の数を分子に，全体で置換した単語数を分母においた比率とした結果が図 4-2 である．ポジティブコーパスの置換結果が図 4-2 中の (a)，ネガティブコーパスの置換結果が (b) である．ポジティブコーパスには，ポジティブな使われ方として置換された（“-p” が末尾に付与された）単語が明らかに多く，逆にネガティブコーパスには，ネガティブな使われ方として置換された（“-n” が末尾に付与された）単語が明らかに多い．ポジティブコーパスの方がポジティブな使われ方のされる単語の多いことは当然考えられることであり，自然な結果になったと言える．

4.4.2 単語の使われ方の判断に関する評価実験の結果

単語の使われ方の判断方法に関する実験結果を表 4-2 に示す．単語の選定方法は提案手法を用いた．また，分散表現のベクトルサイズは 300 に設定し，ニューラルネットワークを判断に用いた実験では負例サンプリングを行った．4.4.1 節と同様に，分割・置換する単語数はポジティブ・ネガティブコーパスに共通する語彙数の 1%，5%，スコアの算出は Accuracy を用いた．この結



(a): ポジティブコーパスにおける
単語の置換結果



(b): ネガティブコーパスにおける
単語の置換結果

図 4-2: 単語の置換結果

表 4-2: 単語の使われ方の判断に関する評価実験の結果

使われ方の判断方法	単語分割数	スコア
ニューラルネットワーク	1%	0.755
ニューラルネットワーク	5%	0.751
ベクトル積	1%	**0.763**
ベクトル積	5%	0.737
分割無し	-	0.748

果から，単語分割数のニューラルネットワークを用いた判断方法より，ベクトル積を用いた判断方法の方が高いスコアを記録したことがわかる．しかし，分割数を増やした際のスコアが，ベクトル積の結果よりも緩やかに低下することを確認した．これは，ニューラルネットワークを用いた判断方法の方が，多くの語彙に対して，より良い使われ方の判断がされたと考えられる．

4.4.3 置換していない単語の感情極性に関する結果

提案手法によって獲得された，極性が付与された単語を基に構築した辞書の概観を考察するため，佐藤ら [5] の手法のうち，SVM のカーネルを RBF カーネルとした実験を行った．なお，分散表現は分割する語彙数を全体の 1%，ベクトルサイズを 300 にして獲得したものに変更している．教師データとして，置換した単語のうち “-p” が付与された単語の正解ラベルを 1，“-n” が付与された単語の正解ラベルを −1 として，入力を前節で獲得した 300 次元の単語分散表現とした．10 分割交差検証を行い，正解率 0.658 を獲得した．その際に獲得した極性情報の例を表 4-3 に示す．多くの単語は直観に反していない結果になった．映画の評価テキストをコーパスとしているため，一部特徴的な極性値が付与された．例として，一般にあまり良くない印象を持つ “dark”（暗い）に対し，ポジティブな極性が付与されている．これは，このコーパスにおいては,「暗い映画館で」と言った使い方や “dark comedy”（一般に扱いにくいものをテーマにした作品の種類）といったあまり一般的ではない使い方が多かったため，ポジティブな印象を持つ語として出力されたと考えられる．逆に，“bad”，“excellent” は直観に反する極性値が付与された．これらの単語は共通して，皮肉としても多く用いられていた．このように文脈によって極性が変化する単語

表 4-3: 獲得した単語極性の例

ポジティブな単語として予測されたもの	ネガティブな単語として予測されたもの
well	hate
success	pity
truth	fake
thrill	rough
dark	common
bad	**excellent**

の代表として，否定と皮肉が考えられる．分割する単語の選定時にこのような単語を考慮できていないことが提案手法の課題として挙げられる．

4.5 結言

本章では，単語を分割・置換したあとのコーパスから獲得された分散表現による文章の極性分類を通し，提案手法の妥当性を示した．実験結果から提案手法にある程度の優位性は認められるが，単語すべてに極性が反転する可能性が存在することを考えると，提案手法の有効性は限定的である．極性分類のための理想的な分散表現においては，極性的な情報以外にも考慮しなくてはならない情報があると考えられる．特に，文脈による単語の使われ方の変化を考慮することを目的にした提案手法において，皮肉に多く用いられている単語を分割することへの優先度は極めて高い．少なくとも分割する単語の選定基準には課題があると考えられる．また，提案手法によって分割された単語を増やすとスコアが落ちるという結果から，一部の単語がより良い分散表現の獲得を妨げていることがわかった．提案手法において，単語の順序を考慮せずに分割していることが大きな影響を与えていると考えられる．少し離れた位置にある単語や，周辺単語の順序に大きな影響を受けるような単語，また皮肉として使われている単語の極性について対応した分割ができていないことが原因と考えられる．

第5章　終わりに

本研究では，コーパスに依存した感情極性辞書を構築するため，単語の極性値をコーパスから自動的に抽出することを試みた．本論文の第1章では，感情極性辞書の自動的な構築の研究背景と本論文の概要について説明した．第2章では，感情極性辞書の構築の関連研究及び本論文で用いた分散表現・文章分類モデルについて説明した．本論文において分散表現の占める重要度は大きく，周辺単語に関する情報を考慮するための大きなファクターとなっている．その関連研究及び本研究における分散表現の利用に関する説明を行った．第3章においては，提案手法の中心たる処理について説明した.「分割する単語の選定基準」「単語の分割・置換手法」が本研究の核をなしており，コーパスから分散表現を獲得することで周辺単語の情報を考慮しやすいことを利用した．具体的には，ある単語の周辺単語が大きく変わる場合にその単語の使われ方も大きく変わる点，ある単語の生起確率は周辺の単語に対応するベクトル積からある程度推察できる点に注目した．これらにより，コーパス内の単語の確率分布を近似的に扱えるようになり，コーパスに依存した単語の分割を可能とした．第4章では，単語を分割・置換した結果の評価及び考察を行った．分割した単語そのものに教師たるデータは存在しないため，分散表現を再度獲得し，それによって評価を行った．理想的な分割手法であれば，すべての単語を適切に分割・置換できるため極性反転が発生しないコーパスを獲得できるという仮定に基づいて評価した．文章分類においては提案手法にある程度の優位性が認められたため，局所的には単語の分割に成功したと言えるが，分割する単語数を増やすとスコアが低下したことから，提案手法は理想的な分割手法とは言えない結果になった．

本実験で分割した単語は必ずコーパスのラベルに依存した極性を持っているため，分割した単語を基にその他の単語の極性値を予測するような発展が容易に可能である．また，英文のポジティブ/ネガティブを例にとり実験を行ったが，本手法は英文以外，また感情極性以外にも適用が可能である．感情極性とは違い人手での評価があまり行われてこなかったラベルについても同様の手法を適用することで，新たな種類のラベル辞書の構築に貢献できると思われる．今後は，単語順序を考慮したうえで感情以外のラベルについても妥当性を調査し，より広い範囲に適用できる手法へ発展させたい．

謝辞

本論文を執筆するにあたり，多大なるご指導を頂いた青山学院大学理工学部情報テクノロジー学科大原剛三教授，ならびに豊田哲也助教に深くお礼申し上げます．また，1 年間共に研究生活を過ごし，多くの教授をいただきました青山学院大学理工学部情報テクノロジー学科発見科学研究室の皆様に感謝します．

参考文献

[1] Wilson, T., Wiebe, J. and Hoffmann, P.: Recognizing contextual polarity in phrase-level sentiment analysis, *Proceedings of the conference on human language technology and empirical methods in natural language processing*, Association for Computational Linguistics, pp. 347–354 (2005).

[2] 高村大也，乾　孝司，奥村　学：スピンモデルによる単語の感情極性抽出，情報処理学会論文誌，Vol. 47, No. 2, pp. 627–637 (2006).

[3] Kamps, J., Marx, M., Mokken, R. J. and De Rijke, M.: Using WordNet to Measure Semantic Orientations of Adjectives., *LREC*, Vol. 4, Citeseer, pp. 1115–1118 (2004).

[4] 中村　拓，乾健太郎：Linked Open Data を利用した単語極性予想によるレビュー感情分析，言語処理学会発表論文集 (2018).

[5] 佐藤貴俊，高村大也，奥村　学：分散表現を用いた単語の感情極性抽出，研究報告自然言語処理 (NL)， Vol. 2016, No. 12, pp. 1–6 (2016).

[6] Kanayama, H. and Nasukawa, T.: Fully automatic lexicon expansion for domain-oriented sentiment analysis, *Proceedings of the 2006 conference on empirical methods in natural language processing*, Association for Computational Linguistics, pp. 355–363 (2006).

[7] 鍜治伸裕，喜連川優:自動構築した評価文コーパスからの評価表現辞書の構築，*DBSJ Letters*, Vol. 6, No. 1 (2007).

[8] 高村大也，乾　孝司，奥村　学：極性反転に対応した評価表現モデル，情報処理学会研究報告自然言語処理 (NL)， Vol. 2005, No. 73 (2005-NL-168), pp. 141–148 (2005).

[9] 張　培楠，小町　守：単語分散表現を用いた多層 Denoising Auto-Encoder による評価極性分類，研究報告自然言語処理 (NL)， Vol. 2015, No. 6, pp. 1–8 (2015).

[10] 中村　拓，田　然，乾健太郎：単語の極性を埋め込んだ分散表現，言語処理学会発表論文集，Vol. 24, pp. 348–351 (2018).

[11] Turney, P. D. and Littman, M. L.: Measuring praise and criticism: Inference of semantic orientation from association, *ACM Transactions on Information Systems (TOIS)*, Vol. 21, No. 4, pp. 315–346 (2003).

[12] 立石健二，石黒義英，福島俊一：インターネットからの評判情報検索，情報処理学会研究報告自然言語処理 (NL)， Vol. 2001, No. 69 (2001-NL-144), pp. 75–82 (2001).

[13] 小林のぞみ，乾健太郎，松本裕治，立石健二，福島俊一：意見抽出のための評価表現の収集，自然言語処理，Vol. 12, No. 3, pp. 203–222 (2005).

[14] 池田大介，高村大也，奥村　学：単語極性反転モデルによる評価文分類，人工知能学会論文誌，Vol. 25, No. 1, pp. 50–57 (2010).

[15] Mikolov, T., Chen, K., Corrado, G. and Dean, J.: Efficient estimation of word representations in vector space, *arXiv preprint arXiv:1301.3781* (2013).

[16] Tai, K. S., Socher, R. and Manning, C. D.: Improved semantic representations from tree-structured long short-term memory networks, *arXiv preprint arXiv:1503.00075* (2015).

[17] Kim, Y.: Convolutional neural networks for sentence classification, *arXiv preprint arXiv:1408.5882* (2014).

[18] Krizhevsky, A., Sutskever, I. and Hinton, G. E.: Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems*, pp. 1097–1105 (2012).

付録

A.1　LMS 上の質問に対する回答

ここでは，青山学院大学理工学部情報テクノロジー学科 2018 年度卒業論文発表会において，LMS を通じて本研究に出された質問とその回答を示す.

Q. コメントです：評価実験の手法として，使用したポジティブコーパスとネガティブコーパスが属しているのと同じドメインの文書をテストデータとして使用しての正解率等の評価が必要だと思います．それによりドメイン間の分類性能の平均値・分散を求め，本手法を用いた場合のドメイン間でのばらつきがどれくらい存在するのかを検定するのが妥当ではないでしょうか．(谷津先生)

A. 評価手法についてのご指摘ありがとうございます．ご指摘の通り，現状の評価方法は，同じドメインの一部をテストデータとして評価する方法をとっております．また，これもご指摘の通り，現状では 1 つのドメインのみを対象とした評価であり，ドメインによる性能の違いを示すには至りませんでした．今後，複数のドメインで同じ評価を行い，ドメイン間の性能のばらつきを調べたいと考えております.