評価表現と文脈一貫性を利用した 教師データ自動生成によるクレーム検出

乾 孝司[†]・梅澤 佑介[†]・山本 幹雄[†]

本論文では、レビュー文書からクレームが記述された文を自動検出する課題に対して、従来から問題となっていた人手負荷を極力軽減することを指向した次の手続きおよび拡張手法を提案する：(1) 評価表現と文脈一貫性に基づく教師データ自動生成の手続き、(2) 自動生成された教師データの特性を踏まえたナীবベイズ・モデルの拡張手法。提案手法では、大量のレビュー生文書の集合と評価表現辞書が準備できれば、クレーム検出規則の作成・維持・管理、あるいは、検出規則を自動学習するために必要となる教師データの作成にかかる人手負荷は全くかからない利点をもつ。評価実験を通して、提案手法によって検出対象文の文脈情報を適切に捉えることで、クレーム文の検出精度を向上させることができること、および、人手によって十分な教師データが作成できない状況においては、提案手法によって大量の教師データを自動生成することで、人手を介在させる場合と同等あるいはそれ以上のクレーム検出精度が達成できることを示した。

キーワード：クレーム、評価表現、文脈一貫性、教師データ

Complaint Sentence Detection via Automatic Training Data Generation using Sentiment Lexicons and Context Coherence

TAKASHI INUI[†], YUSUKE UMESAWA[†] and MIKIO YAMAMOTO[†]

We propose an automatic method for detecting complaint sentences from review documents. The proposed method consists of two procedures. One is a data generation procedure using sentiment lexicons and context coherence and the other is the expansion of a naive Bayes classifier based on the characteristics of the training data. This method has an advantage of not requiring human effort for the creation of large-scale training data and management of rules for complaint detection. The experimental results indicate that this method is more effective than the baseline methods.

Key Words: *complaint, sentiment expression, context coherence, labeled data*

[†] 筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻, Department of Computer Science, Graduate school of SIE, University of Tsukuba

1 はじめに

インターネットの普及により、個人がWeb上で様々な商品を購入したり、サービスの提供を受けることが可能になった。また、これに伴い、商品やサービスに対する意見や感想が、大量にWeb上に蓄積されるようになった。これらの意見や感想は、ユーザが商品やサービスを購入する際の参考にするだけでなく、企業にとっても商品やサービスの改善を検討したり、マーケティング活動に活用するなど、利用価値の高い情報源として広く認識されている。

近年ではさらに、ユーザ参加型の商品開発が注目されるなど、ユーザと企業とがマイクロブログやレビューサイト等のソーシャルメディアを通して、手軽に相互にコミュニケーションを持つことも可能となっている。そして、このようなコミュニケーションの場においては、いわゆる「クレーム」と呼ばれる類のユーザの意見に対して企業側は特に敏感になる必要があり、ユーザが発言したクレームに対しては、適切に対応することが望まれている。しかしながら、このようなコミュニケーションの場では、次のような理由からユーザのクレームを見落としてしまう懸念がある。

- 見落とし例 1：特に、マイクロブログ型サービスを通じたコミュニケーションでは、多対一型のコミュニケーション、つまり、大勢のユーザに対して少数の企業内担当者が同時並行的にコミュニケーションを持つことが多く、そのため、一部のユーザが発言したクレームを見落としてしまう可能性がある。
- 見落とし例 2：特に、レビューサイトを通じたコミュニケーションでは、ユーザは様々な意見をひとつのレビュー文書中に書き込むことが多く、その中に部分的にクレームが埋め込まれることがある（図 1 および図 2 に例を示す。下線部がクレームを示す）。この場合、レビューの中からクレームを見つける必要があるが、これらの一部を見落としてしまう可能性がある。

本論文では、上記のうち、2 つ目の見落とし問題に対処すべく、レビューからクレームを自動

従業員の方は親切でした。最寄りの病院など教えて頂き、とても助かりました。ありがとうございました。ただ残念だったのが、シャワーの使い方がよくわからなかったことです。使い方の説明をおいて頂きたいです。

図 1 クレームが含まれたレビューの例 1

ひさびさの家族旅行のときに利用しました。せっかくの家族旅行なのにお風呂の汚れが目立っていたことが気になってしかたがありませんでした。他はだいたい満足です。

図 2 クレームが含まれたレビューの例 2

検出する手法について述べる。より具体的には、まず、文単位の処理を考え、表 1 のような内容を含む文を「クレーム文」と定義する。そして、レビューが入力された際に、そのレビュー中の各文のそれぞれに対して、それらがクレーム文かそうでないかを自動判定する手法について検討する。

これまで、テキストからクレームを検出することを目的とした先行研究としては、永井らの研究(永井, 高山, 鈴木 2002; 永井, 増塩, 高山, 鈴木 2003)がある。永井らは、単語の出現パターンを考慮した検出規則に基づいたクレーム検出手法を提案している。しかしながら、彼らの手法のように人手で網羅的に検出規則を作成するには、作成者がクレームの記述のされ方に関する幅広い言語的知識を有している必要がある。また、現実的に検出規則によって運用するには、膨大な量の規則を人手で作成・維持・管理する必要がある。人的負荷が高いという問題がある。この問題に対する解決策のひとつとして、教師あり学習によって規則を自動学習することが考えられるが、その場合でも、事前に教師データを準備する必要がある。単純には、教師データの作成に労力を要するという別な問題が発生してしまう。

本論文では、上記のような背景を踏まえて、人的な負荷をなるべく抑えたクレーム検出手法を提案する。より具体的には、レビュー文書からクレーム文を自動検出する際の基本的な設定として、テキスト分類において標準的に利用されるナイーブベイズ・モデルを適用することを考え、この設定に対して、極力人手の負荷を軽減させるために、次の手続きおよび拡張手法を提案する。

- 評価表現および文脈一貫性に基づく教師データ自動生成手法を提案する。従来、学習用の教師データを作成するには負荷の高い人手作業に頼らざるを得なかったが、本研究では既存の言語資源と既存の知見に基づくことで、人手作業に頼らずに教師データを自動生成する手法を提案する。
- 次に、上記で生成された教師データに適したモデルとなるように拡張されたナイーブベイズ・モデルを提案する。上記の提案手法によって生成された教師データは自動化の代償として人手作成されたデータと比べて質が劣化せざるを得ず、標準的な分類モデルをそのまま適用するだけでは期待した精度は得られない。本研究では上記のデータ生成手

表 1 クレーム文の定義

クレーム	苦情	意味	商品やサービスに対して不満な感情や不満な点をあらわしている
		例文 1	最初に案内された部屋がカビ臭かったので残念でした。
		例文 2	何故同じ室料でこのような違いが有るのでしょうか？
	要求	意味	商品やサービスに対して不満な点を解消するための要望を述べている
		例文 1	リムジンバスも 2 階を乗降場にして下さい。
		例文 2	あんなに遠いなら送迎ぐらいしてほしい。

法で生成されるデータが持つ特性を踏まえて、ナイーブベイズ・モデルを拡張する。

提案手法では、従来手法で問題となっていた検出規則の作成・維持・管理、あるいは、規則を自動学習するために必要となる教師データの作成にかかる人手負荷は全くかからない利点をもつ。本論文では、上記の手続きおよび拡張手法について、実データを用いた評価実験を通して、その有効性を検証する。本論文の構成は以下の通りである。まず2節で教師データの自動生成手法について説明する。その後、3節でナイーブベイズ・モデルの拡張について説明し、4節で評価実験について述べる。5節で関連研究を整理した後、6節で本論文をまとめる。

2 教師データ自動生成

2.1 教師データ

まず、生成したい教師データについて整理する。本研究では、クレーム文を検知するためにナイーブベイズ分類器 (Manning and Schütze 1999) を構築し、文がクレームを表しているか、あるいはクレームを表していないかのどちらかに分類したい。このような分類器の構築に必要な教師データは、言うまでもなく、クレームを表している文（以下、クレーム文と呼ぶ）の集合と、クレームを表していない文（以下、非クレーム文）の集合となる。以下では、説明の便宜上、このデータ集合を得る手続きをラベル付けと呼び、【クレーム】および【非クレーム】というラベルによって、どちらの集合の要素となるかを区別することとする。例えば、図1の各文に対してラベル付けが実施されたとすると、次のようなラベル付きの教師データが得られる。

- 【非クレーム】従業員の方は親切でした。
- 【非クレーム】最寄りの病院など教えて頂き、とても助かりました。
- 【非クレーム】ありがとうございました。
- 【クレーム】ただ残念だったのが、シャワーの使い方がよくわからなかったことです。
- 【クレーム】使い方の説明をおいて頂きたいです。

本論文で提案する教師データ自動生成手法は、評価表現の情報に基づくラベル付けステップと、ある文の文脈に対する文脈一貫性の情報に基づくラベル付けステップの2ステップで構成される。各ステップをそれぞれ核文ラベル付けおよび近接文ラベル付けと呼ぶことにし、以下で順に説明する。

2.2 核文ラベル付け

核文ラベル付けは、評価表現の情報に基づいて行う。評価表現とは、「おいしい」や「まずい」等、評価対象に対する評価を明示的にあらわす言語表現のことである。一般的には、これら表現に「おいしい／肯定」や「まずい／否定」のような肯定・否定の評価極性値を付随させたものを集めて評価表現辞書と呼ばれている (乾, 奥村 2006)。

核文ラベル付けステップでは、評価表現辞書に否定極性として登録されている評価表現に着目し、このような評価表現を含む文はクレームを表しやすいと仮定する。そして、否定極性の評価表現を含む文をクレーム文としてラベル付けする。以降、この手続きで得られる文を次ステップで得られる文と区別するため核文と呼び、特に、核文がクレーム文である場合はクレーム核文と呼ぶ。例えば、「まずい／否定」という単語が評価表現辞書に登録されている場合、次の例文はクレーム核文としてラベル付けされる。

- 【クレーム（核）】朝食のカレーがまずい。

もし、ある文が肯定極性をもつ評価表現を含み、かつ「ない」や「にくい」などの否定辞が評価表現の3単語以内に後続していた場合もクレーム核文としてラベル付けする。例えば、次の例文は「おいしい／肯定」の直後に否定辞「ない」が後続しているため、クレーム核文としてラベル付けされる。

- 【クレーム（核）】朝食のカレーがおいしく ない。

また、評価表現の否定極性と肯定極性を読み替えて上記と同様の手続きを行った場合に得られる文を非クレーム核文と呼び、クレーム核文と同じようにラベル付けしておく。

- 【非クレーム（核）】朝食のカレーがおいしい。
- 【非クレーム（核）】ハヤシライスは別にまずく はない。

さらに、「ほしい」等の要求表現を集めた要求表現辞書が利用できる場合は、次の例のように要求表現を含む文をクレーム核文としてラベル付けする¹。ただし、要求表現に注目したラベル付けの場合は、評価表現の時とは違って、否定辞の有無に関係なくクレーム核文としてラベル付けする。

- 【クレーム（核）】朝食に和食メニューをもっと増やしてほしい。
- 【クレーム（核）】朝食を洋風なものばかりにしてほしく ない。

以降、クレーム核文と非クレーム核文をあわせた文の集合を S_{core} であらわす。

2.3 近接文ラベル付け

那須川ら(那須川, 金山 2004)は、彼らの論文の中で、評価表現の(文をまたいだ)周辺文脈には以下のような傾向があると述べており、これを評価表現の文脈一貫性と呼んだ。

文書中に評価表現が存在すると、その周囲に評価表現の連続する文脈(以降、評価文脈²)が形成されることが多く、その中では、明示されない限り、好不評の極性が一致する傾向がある。

¹ この操作によって、例えば、「このサービスは是非今後も継続してほしい」というような肯定的な要求については誤ってクレーム文として扱ってしまう。しかし、後述する本研究で使ったデータセットでは、上記のような事例はごく稀であり、本研究ではこのような肯定的な要求に対する特別な処理は施していない。

² 元論文では「評価部脈」と書かれているが、これは「評価文脈」の書き誤りであると考えられる。

本研究では、この評価表現の文脈一貫性の考え方に基いて近接文ラベル付けを行う。先の核文ラベル付けの際に考慮した評価表現（あるいは要求表現）を含む文の周辺文脈について、「評価表現（要求表現）の存在に基づいて（非）クレーム文として選ばれた文の前後文脈に位置する文は、やはり（非）クレーム文である」という仮定をおき、この仮定に従って、核文の周辺文脈に対してラベル付けを行う。この手続きで得られる文を近接文（より詳細にはクレーム近接文あるいは非クレーム近接文）と呼ぶ。

近接文ラベル付けの手続きを Algorithm 1 に示す。この手続きへの入力は、核文ラベル付けを終えたレビュー d と、核文に対する周辺文脈の長さを決定する窓枠長 N (≥ 0) であり、レビュー d に含まれる核文に対して、レビューの先頭側に現れる核文から末尾側に現れる核文に向かって順に処理が進む。なお、Algorithm 1 において、 s_i はレビュー d 内の先頭から i 番目の

Algorithm 1 近接文ラベル付け

```

INPUT   $d$ : 核文ラベル付けを終えたレビュー,  $N$ : 周辺文脈の窓枠長
1: set label_array[ $|d|$ ][ $|d|$ ] // 処理過程のラベル情報を格納する配列を用意
2: for  $i = 1$  to  $|d|$  do
3:   if  $s_i$  が核文である then
4:     foreach  $s$  in  $s_i$  に対して  $d$  の先頭側に近接している  $N$  個の文の集合
5:       if  $s$  あるいは、 $s$  と  $s_i$  の間にある文が核文でない then BackwardLabeling( $s_i, s$ )
6:       foreach  $s$  in  $s_i$  に対して  $d$  の末尾側に近接している  $N$  個の文の集合
7:         if  $s$  あるいは、 $s$  と  $s_i$  の間にある文が核文でない then ForwardLabeling( $s_i, s$ )
8:     end if
9:   end for
10: for  $i = 1$  to  $|d|$  do
11:   if label_array[ $s_i$ ] の要素が全て “クレーム” then
12:      $s_i$  をクレーム近接文としてラベル付けする
13:   elseif label_array[ $s_i$ ] の要素が全て “非クレーム”
14:      $s_i$  を非クレーム近接文としてラベル付けする
15:   end if
16: end for
17: def BackwardLabeling( $s_i, s$ )
18:   if  $s_i$  あるいは、 $s$  と  $s_i$  の間にある文が “しかし” 等の逆接関係の接続詞を含まない then
19:     if  $s_i$  はクレーム核文である then push(label_array[ $s$ ], “クレーム”)
20:     else push(label_array[ $s$ ], “非クレーム”)
21:   else
22:     if  $s_i$  はクレーム核文である then push(label_array[ $s$ ], “非クレーム”)
23:     else push(label_array[ $s$ ], “クレーム”)
24:   end if
25: end def
26: def ForwardLabeling( $s_i, s$ )
27:   if  $s$  あるいは、 $s$  と  $s_i$  の間にある文が “しかし” 等の逆接関係の接続詞を含まない then
28:     if  $s_i$  はクレーム核文である then push(label_array[ $s$ ], “クレーム”)
29:     else push(label_array[ $s$ ], “非クレーム”)
30:   else
31:     if  $s_i$  はクレーム核文である then push(label_array[ $s$ ], “非クレーム”)
32:     else push(label_array[ $s$ ], “クレーム”)
33:   end if
34: end def

```

文をあらわし, $|d|$ は d 内の文数をあらわす. 処理の大きな流れとしては, line.2-9 でラベル付けされる文が選択され, line.10-16 でラベル付けが実施される. line.17-34 の各関数では, 付与するラベルの種類 (“クレーム” か “非クレーム”) を確定する際に必要な仮のラベル情報が決められ, その情報が格納される.

図 3 を使って近接文ラベル付けの具体的な実行例を示す. 図の例では, 対象となるレビューは 8 つの文から構成されており, 核文ラベル付けによって文 s_1 が非クレーム核文, 文 s_5 がクレーム核文とラベル付けされた状態であり, この状態から近接文ラベル付けが開始される. 窓枠長は $N = 2$ とする. この場合, まず, 核文 s_1 の周辺文脈に対する処理がなされる (Algorithm 1 の line.3). s_1 は文書の先頭文であり前方文脈 (Backward context) は存在しない. そのため, 後方文脈 (Forward context) の s_2 と s_3 に対してのみ処理がなされ (line.6-7), それぞれ “非クレーム” ラベルが配列に格納される (line.29). 次に核文 s_5 の周辺文脈に対する処理がなされる. s_5 はクレーム文であるため, 前方文脈では s_3 に対して 2 つ目のラベル “クレーム” が格納され (line.19), また新たに s_4 に対して “クレーム” ラベルが格納される (line.19). 後方文脈では, まず s_6 に対して “クレーム” ラベルが配列に格納される (line.28). その一方で, s_7 は逆接関係の接続詞「しかし」の影響があるため, “クレーム” ではなく “非クレーム” ラベルが格納される (line.31). 最後に, 各文に対して格納されたラベル情報をチェックし, 格納されたラベルに不整合がない場合は, そのラベル情報に従ってラベル付けを行う (line.10-16). 不整合が生じている場合はその文に対してどのラベルも付与しない. 以上の操作によって, この例では 2 つの核文

図 3 近接文ラベル付けの例 (窓枠長 $N = 2$ の場合)

から新たに 4 つの近接文 (s_2, s_4, s_6 および s_7) が得られる.

以降, クレーム近接文と非クレーム近接文をあわせた文の集合を $\mathcal{S}_{sat}$ ³であらわす, また, 必要に応じて, 核文の前方文脈から得られた近接文 $\mathcal{S}_{sat}^B$ と, 後方文脈から得られた近接文 $\mathcal{S}_{sat}^F$ を区別する ($\mathcal{S}_{sat} = \mathcal{S}_{sat}^B \cup \mathcal{S}_{sat}^F$).

3 ナイーブベイズ・モデルの拡張

3.1 ナイーブベイズ・モデル (Naïve Bayes model; NB)

前節で述べた手法によって自動生成された教師データは, 人手によって作成された教師データと比べて質が劣化せざるを得ず, 標準的な分類モデルをそのまま適用するだけでは期待した精度は得られない. そこで, 前節で述べた手法で得られる劣化を含むデータを使用するという前提をおき, この劣化データがもつ特性を踏まえてナイーブベイズ・モデルを拡張することを考える. 以下では, まず, 通常のナイーブベイズ・モデル (多項モデル) について述べ, その後, モデルの拡張について述べる.

ナイーブベイズ分類器では, ある文 s の分類クラスを判定する際に, 条件付き確率 $P(c|s)$ を考え, この確率値が最大となるクラス $\hat{c}$ を分類結果として出力する. つまり,

$$\hat{c} = \arg \max_c P(c|s) \quad (1)$$

である. 通常のナイーブベイズ・モデルでは上式を次のように展開する.

$$\begin{aligned} \arg \max_c P(c|s) &= \arg \max_c P(c)P(s|c) \\ &= \arg \max_c \left\{ \log p_c + \sum_{w \in \mathcal{V}} n_w(s) \log q_{w,c} \right\} \end{aligned} \quad (2)$$

ここで, $\mathcal{V}$ は語彙集合, $n_w(s)$ は文 s における単語 w の出現回数をあらわす. また, $q_{w,c}$, p_c は教師データを使ってそれぞれ以下の式で計算される. 本研究ではパラメータを推定する際にラプラススムージング (Manning and Schütze 1999) を用いる.

$$q_{w,c} = \frac{n_{w,c}(\mathcal{D}) + 1}{\sum_w n_{w,c}(\mathcal{D}) + |\mathcal{V}|} \quad (3)$$

$$p_c = \frac{n_c(\mathcal{D}) + 1}{\sum_c n_c(\mathcal{D}) + |\mathcal{C}|} \quad (4)$$

³ 添字 sat は satellite の略である.

ここで, $\mathcal{D}$ は教師データとなる文の集合, $n_{w,c}(\mathcal{D})$ はデータ $\mathcal{D}$ においてクラス c に属する文に現れる w の出現回数, $n_c(\mathcal{D})$ はデータ $\mathcal{D}$ においてクラス c に属する文の数, $|\mathcal{V}|$ は語彙の種類数, $|\mathcal{C}|$ は分類クラスの種類数である.

以上からもわかるように, 通常モデルでは, 分類対象となる文内の情報のみを考慮し, 分類対象文の周辺文脈の様子は全く考慮されない. たとえ同一文書内であっても個々の文は独立に評価・分類する. また, 教師データの利用にあたって, 当然のことながら, 核文であるか近接文であるかといった区別はなく, 両タイプの文が同等にモデルの構築に利用される.

3.2 モデル拡張

前節で述べた教師データ生成過程から得られるデータには, 核文および近接文という 2 種類の文が存在する. この 2 種類の文のうち, 核文は近接文とは独立にラベル付けされる一方で, 近接文は核文の情報に基いて間接的にラベル付けされる. そのため, 核文ラベル付けが結果として誤りであった事例に関しては近接文もその誤りの影響を直接受けることになる. また, 当然ながら, 文脈一貫性の仮定が成立しない事例もあり得る. このような理由から, 近接文は核文に比べて相対的に信頼性の低いデータとなる可能性が高い.

そこで, このことを考慮し, 核文と近接文の情報をモデル内で区別して扱い, 近接文の情報がモデル内で与える影響を下げるよう, 式 (2) の代わりに次のような式 (5) を用いる.

$$\begin{aligned} \arg \max_c P(c|s) &= \arg \max_c P(c)P(s|c) \\ &= \arg \max_c \left\{ \log p_c + \sum_w n_w(s) \log q_{w,c}^{tgt} \right. \\ &\quad \left. + \frac{1}{|ctx(s, N)|} \sum_w n_w(ctx(s, N)) \log q_{w,c}^{ctx} \right\} \end{aligned} \quad (5)$$

右辺第 3 項に現れる $ctx(s, N)$ は s の周辺文脈に位置する前方および後方のそれぞれ N 文から構成される文の集合を表しており, この項が分類対象の周辺文脈をモデル化している. この項の係数 $1/|ctx(s, N)|$ で, 周辺文脈の文数に応じてその影響を調整している. なお, $n_w(ctx(s, N))$ は, 集合 $ctx(s, N)$ の要素となる全ての文における単語 w の総出現回数をあらわす. また, 右辺第 2 項は通常モデルと同様に分類対象となる文をモデル化したものであるが, 第 3 項の周辺文脈との区別を明瞭にするため, $q_{w,c}^{tgt}$ という記号を新たに導入した⁴.

式 (5) の $q_{w,c}^{tgt}$ と $q_{w,c}^{ctx}$, および p_c はそれぞれ次式で求める. 式中の各記号の意味は式 (3), 式 (4) と同様である. ここで, $\mathcal{D}_{tgt}$ は分類対象文をモデル化するための教師データ集合, $\mathcal{D}_{ctx}$ は分類対象の周辺文脈をモデル化するための教師データ集合である. 基本的には, 前節で得られる教

⁴ 上付きの添字 tgt と ctx は, それぞれ target, context をあらわしている.

師データのうち、核文データを $\mathcal{D}_{tgt}$ に割り当て、近接文データを $\mathcal{D}_{ctx}$ に割り当てて、正確な記述は後述の 3.3 節で与える。

$$q_{w,c}^{tgt} = \frac{n_{w,c}(\mathcal{D}_{tgt}) + 1}{\sum_w n_{w,c}(\mathcal{D}_{tgt}) + |\mathcal{V}_{tgt}|} \quad (6)$$

$$q_{w,c}^{ctx} = \frac{n_{w,c}(\mathcal{D}_{ctx}) + 1}{\sum_w n_{w,c}(\mathcal{D}_{ctx}) + |\mathcal{V}_{ctx}|} \quad (7)$$

$$p_c = \frac{n_c(\mathcal{D}_{tgt}) + 1}{\sum_c n_c(\mathcal{D}_{tgt}) + |\mathcal{C}|} \quad (8)$$

以降、便宜的にこの拡張されたモデルを **NB+ctx** と呼ぶ。

さらに、式 (5) の第 3 項について、周辺文脈を分類対象文からの相対位置で詳細化した

$$\frac{1}{|ctx(s, N)|} \left\{ \sum_w n_w(Bctx(s, N)) \log q_{w,c}^{Bctx} + \sum_w n_w(Fctx(s, N)) \log q_{w,c}^{Fctx} \right\} \quad (9)$$

を代わりに利用するモデルも考えられる。ここで、 $Bctx(s, N)$ は、 s の前方文脈に位置する N 文から構成される文の集合であり、 $Fctx(s, N)$ は同様に後方文脈で構成される文集合である。また、式中の $q_{w,c}^{Bctx}$ および $q_{w,c}^{Fctx}$ は次式で求める。ただし、 $\mathcal{D}_{ctx} = \mathcal{D}_{ctx}^B \cup \mathcal{D}_{ctx}^F$ である。

$$q_{w,c}^{Bctx} = \frac{n_{w,c}(\mathcal{D}_{ctx}^B) + 1}{\sum_w n_{w,c}(\mathcal{D}_{ctx}^B) + |\mathcal{V}_{Bctx}|} \quad (10)$$

$$q_{w,c}^{Fctx} = \frac{n_{w,c}(\mathcal{D}_{ctx}^F) + 1}{\sum_w n_{w,c}(\mathcal{D}_{ctx}^F) + |\mathcal{V}_{Fctx}|} \quad (11)$$

以降、便宜的にこのモデルを **NB+ctxBF** と呼ぶ。

3.3 データ割当規則

ここでは、さきほどの説明で保留していた、パラメータ推定の際に必要な教師データの与え方について述べる。ここで、前節で述べた手法によって得られるデータ集合を確認すると、

- $\mathcal{S}_{core}$: クレーム核文と非クレーム核文をあわせた文の集合
- $\mathcal{S}_{sat}$: クレーム近接文と非クレーム近接文をあわせた文の集合
- $\mathcal{S}_{sat}^B$: $\mathcal{S}_{sat}$ の要素のうち、核文の前方文脈から得られた文で構成される集合
- $\mathcal{S}_{sat}^F$: $\mathcal{S}_{sat}$ の要素のうち、核文の後方文脈から得られた文で構成される集合

であり, $\mathcal{S}_{sat} = \mathcal{S}_{sat}^B \cup \mathcal{S}_{sat}^F$ であった.

これらデータ集合に対して, まず, 核文と近接文を区別しない単純な割当として, 得られた全データをまとめて利用することが考えられる. この場合, 拡張モデル NB+ctx における割当は,

- $\mathcal{D}_{tgt} = \mathcal{S}_{core} \cup \mathcal{S}_{sat}$
- $\mathcal{D}_{ctx} = \mathcal{S}_{core} \cup \mathcal{S}_{sat}$

となる. これを以降 **NB+ctx(all)** と呼ぶ. また同様に, 拡張モデル NB+ctxBF における単純な割当は,

- $\mathcal{D}_{tgt} = \mathcal{S}_{core} \cup \mathcal{S}_{sat}$
- $\mathcal{D}_{ctx}^B = \mathcal{S}_{core} \cup \mathcal{S}_{sat}$
- $\mathcal{D}_{ctx}^F = \mathcal{S}_{core} \cup \mathcal{S}_{sat}$

となるが, これは先の NB+ctx(all) と事実上同等となるため以降の議論では割愛する.

次に, 核文と近接文の区別を考慮したデータ割当を考える. 拡張モデル NB+ctx における割当としては,

- $\mathcal{D}_{tgt} = \mathcal{S}_{core}$
- $\mathcal{D}_{ctx} = \mathcal{S}_{sat}$

が考えられる. これを以降 **NB+ctx(divide)** と呼ぶ. また同様に, 拡張モデル NB+ctxBF におけるデータ割当として,

- $\mathcal{D}_{tgt} = \mathcal{S}_{core}$
- $\mathcal{D}_{ctx}^B = \mathcal{S}_{sat}^B$
- $\mathcal{D}_{ctx}^F = \mathcal{S}_{sat}^F$

が考えられる. これを以降 **NB+BFctx(divide)** と呼ぶ.

最後に, 通常のナイーブベイズについて考えると, この場合は, もともとのモデルにデータを区別する枠組みが存在しないため,

- $\mathcal{D} = \mathcal{S}_{core} \cup \mathcal{S}_{sat}$

という割当のみを考えることになる. なお, $\mathcal{D} = \mathcal{S}_{core}$ という核文のみを考慮し, 近接文を利用しない割当も考えられるが, これは $\mathcal{D} = \mathcal{S}_{core} \cup \mathcal{S}_{sat}$ において近接文の窓枠長を 0 とする場合に等しいため, 割当規則として明示的には議論しないが, 第 4 節の評価実験では, 近接文の窓枠長が 0 の場合も含めて議論する. 以降, これを **NB** と呼ぶ.

ここまでの議論を整理すると, 前節の手法で自動生成された教師データを利用するという前提のもとで, 通常のナイーブベイズ・モデルも含めて, 4 つのクレーム文検出モデルが与えられたことになる. 次節では, 評価実験を通じて, これらの有効性を検証していく.

4 評価実験

4.1 検証項目

評価実験を通して、提案手法の有効性を検証する．具体的には以下の 3 項目を検証する．

- (1) 提案手法の比較：前節までで述べた 4 つのクレーム文検出モデルの中で、どのモデルが最良であるかを検証する．
- (2) 他手法との比較：提案手法と他手法との比較実験を行い、その結果から提案手法の有効性を検証する．
- (3) 学習データ量とクレーム文検出精度の関係について：提案したデータ生成手法は学習データを自動生成できるため、人手による生成に比べて遥かに多くの教師データを準備できる．この利点を実験を通して検証する．

4.2 実験の設定

実験には、楽天データ公開⁵において公開された楽天トラベルの施設データを利用した．このデータは約 35 万件（平均 4.5 文／件）の宿泊施設に関するレビューから構成されており、ここから、無作為に選んだ 1,000 レビューに含まれる文を評価用データとして用い、残りを教師データ生成用に利用した．評価用データには 4,308 文が含まれており、その内の 24%にあたる 1,030 文がクレーム文であった．つまり、4,308 文からクレームを述べている 1,030 文を過不足なく検出することがここでの実験課題である．評価用データの作成では、まず、レビュー文書中の各文が 1 行 1 文となるようにデータを整形し、それを作業者に提示した．そして作業者は、与えられたデータの 1 文（1 行）ごとにクレーム文か否かを判定していった．なお、ある文の判定時には、同一レビュー内の他の全ての文が参照できる状態になっている．2 名の作業者によって上記の作業を独立に並行に行ったが、このうち 1 名の作業結果を評価用データとして採用した．2 名の作業者間の一致度を κ 係数の値によって評価したところ、 $\kappa = 0.93$ であった．この結果は、作業者間の判断が十分に一致していたことを示している．作業者間で判断が一致しなかった事例としては、文が長く、ひとつの文で複数の事柄が述べられている場合や、「長身で据え置きのものでは短くて…」のように、クレームの原因が宿泊施設側にあるとは必ずしも言えない場合が多かった．

教師データ生成時に必要となる評価表現辞書には、高村ら（高村，乾，奥村 2006）の辞書作成手法に基いて作成された辞書を使用した．ただし、高村らのオリジナルの辞書は自動構築されたもので、そのままでは誤りが含まれているため、以下の手続きによって誤り修正を施し、本実験で使用する辞書として採用した．オリジナルの辞書には各登録語に対して肯定／否定の強

⁵ <http://rit.rakuten.co.jp/rdr/index.html>

さを示すと解釈できる $[-1, 1]$ の範囲のスコアが付与されている。このスコアは、値が大きいほど肯定、また、小さいほど否定をあらわし、0 付近はどちらでもないことを示していると解釈できる。そこでまず、このスコアの絶対値の大きいものから 0.9 付近までの単語を自動的に選択した。そして、選択された各単語に対して人手による誤り修正を施し、結果として肯定表現 760 件、否定表現 862 件からなる辞書を作成し、本実験に用いた。

また、要求表現辞書として、「欲しい」、「ほしい」、「べし」からなる辞書を作成して実験に使用した⁶。周辺文脈の窓枠長 N の指定は、データ作成時、モデル学習時、評価用データの分類時のすべての過程で同期させている。また、各データの単語分割は MeCab⁷によって行った。また、計算の都合上、 $N = 0$ の場合は $1/|ctx(s, N)| = 0$ とした。

今回のように、分類すべきクラスがクレーム／非クレームという 2 クラスの分類問題の場合、式 (1) による意思決定は、以下の式 (12) の符号が正の場合にクレームと判定することになる。

$$P(\text{“クレーム”}|s) - P(\text{“非クレーム”}|s) \quad (12)$$

しかし、本研究では、式 (12) に意思決定の閾値 θ を加えた次の条件式を新たに導入し、この条件式が成立する場合にクレームと判定し、成立しない場合は非クレームと判定することとした。

$$P(\text{“クレーム”}|s) - P(\text{“非クレーム”}|s) > \theta \quad (13)$$

式 (13) の左辺は、クレームと判定する際の確信度を示していると考えることができ、閾値 θ はこの確信度に応じて出力を制御する役割を持つ。閾値を $\theta = 0$ と設定すると、これは式 (12) を用いた通常の意味決定と同じ動作となる。閾値を 0 から大きくすると、より確信度が高い場合のみクレームと判定することになる。実験では、閾値 θ を増減させ、以下の式で計算される適合率および再現率、あるいはその要約である 11 点平均適合率 (Manning, Raghavan, and Schütze 2008) を求め、検出精度を評価した。11 点平均適合率とは再現率が $\{0.0, 0.1, \dots, 1.0\}$ となる 11 点における適合率の平均値である。

$$\text{適合率} = \frac{\text{正しくクレーム文として検出できた数}}{\text{クレーム文として出力された数}} \quad (14)$$

$$\text{再現率} = \frac{\text{正しくクレーム文として検出できた数}}{\text{クレーム文の数}} \quad (15)$$

データにおけるクレーム文と非クレーム文の割合等に応じて、検出性能に対して最適な θ を自動推定することも考えられるが、これについては今後の課題である。

⁶ 評価表現辞書と比べて要求表現辞書への登録単語数が少ない印象を受けるかもしれない。しかし、要求表現は評価表現とは違い、「～してほしい」のように自立語に付随する形態を取りやすく、そのため辞書に登録できる単語もそれほど多くない。

⁷ <http://mecab.sourceforge.net/>

4.3 提案手法の比較

実験結果を図4に示す。このグラフは、4つの各検出モデルについて、考慮する周辺文脈の窓枠長を変化させながら性能変化をプロットしたものである。文脈長 $N = 0$ の場合は、どのモデルも同じになるため、グラフ上では1点に集まっている。

NBモデルの結果(“◇”)を基準に考えると、核文と近接文を区別しないNB+ctx(all)では文脈長を $N = 0$ から $N \geq 1$ のどの文脈の長さに変更しても性能が向上しない一方で、核文と近接文の区別を考慮するNB+ctx(divide)とNB+BFctx(divide)は文脈長を $N \geq 1$ にすることで、一貫して性能が向上することがわかる。このことから、文脈情報を適切にモデルに反映させるためには、単にモデルを拡張するだけでは効果がなく、データとモデルを上手く組合せて、核文と近接文を区別することが重要であることが確認できる。性能の向上が見られたNB+ctx(divide)とNB+BFctx(divide)を比較すると、どちらも $N \geq 1$ の場合は文脈長の変化に対しては鈍感な傾向を示しているが、近接文の相対位置を考慮するNB+BFctx(divide)の方が総じて良い結果を示しており、本論文で述べた4つのクレーム文検出モデルの中では、NB+BFctx(divide)モデルが最良であることがわかる。

次に、教師データとして自動生成されたクレーム近接文に含まれる単語を確認したところ、表2のような単語がクレーム核文には現れず、クレーム近接文にのみ現れていた。このような

図4 実験結果 (提案手法の性能比較)

表 2 クレーム近接文にのみ現れていた単語

「むせ返る」「激臭」「ばらばら」「不遜」
「投げつける」「飛び上がり」「黒焦げ」「むかつく」

単語の情報は、周辺文脈の情報を取り込むことで初めて考慮できるようになった情報であり、定性的にもクレーム検出における周辺文脈情報の利用の有効性が確認できる。

また文脈情報を取り込むことで正しく分類ができるようになった事例を以下に示す。下線の引かれた文が分類対象であり、左端の数字は分類対象文からの相対位置を示す。

- 【正しくクレーム文であると判定できた例】
 - 2: 疲れていたので苦情をいうのも面倒で、さっさとチェックアウトしました。
 - 1: 普段だったらクレームを入れるレベルです。
 - 0: もう宿泊はないと思います。
 - +1: 残念です。
- 【正しく非クレーム文であると判定できた例】
 - 1: 昨年寒い1月に宿泊した際、犬を入室させてもらえ助かりました。
 - 0: 今回は寒くはないが飼い犬のミニチュアダックスが老齢18歳で泊めて頂け大変助かりました。
 - +1: また泊めて頂きます。
 - +2: 感謝。

次に、誤りの傾向を分析したところ、以下のような事例について、判定誤りが多く見られた。

- 【誤ってクレーム文と判定する例】
 - A. 不満の表明ではあるが、その対象・原因が宿泊者にある場合
 - * 【例】仕事で到着が遅くなり、ゆっくりできなかったのが残念でした。
 - B. クレームの対象となりやすい事物が文中に多く記述されている場合
 - * 【例】部屋はデスク、姿見、椅子、コンセントがあり、…従業員の対応もまずまずでした。
- 【誤って非クレーム文と判定する例】
 - C. 記述の省略を伴う場合
 - * 【例】バイキングにステーキがあればなあ。
 - D. 外部的な知識を要する場合
 - * 【例】全体的な評価としてはEランクでした。

誤ってクレーム文と判定する事例のうち、A.のような事例に対応するには、意見の対象や原因を特定する等の詳細な自動解析の実現が望まれる。手元のデータによると、B.に該当する上述の例のうち、下線部がクレーム対象となりやすい事物であった。このような事例については、

文中では名詞が多く現れることから、単語の品詞情報を考慮する等、単語や単語クラス毎にモデル内での扱いを変更することが考えられる。また、誤って非クレーム文と判定する事例については、「E ランク」を否定極性の単語として扱うなど、ヒューリスティック規則によるチューニングは可能であるが、総体的には現在の技術では改善が困難な事例が多い印象である。

4.4 他手法との比較

次に、提案手法と他手法との比較実験を行い、その結果から提案手法の有効性を検証する。他手法としては、以下に示す 3 手法を検討した。始めの 2 つは、従来から考えられるラベル付け方法に基づく手法であり、残りの 1 つは、教師あり学習を適用しない、辞書の情報に基づいたルールベースの手法である。

- 人手によって教師データを作成する手法（以下、人手ラベル）

教師データ用のレビュー集合から 2,000 件のレビューを無作為に抽出し、そこに含まれる全ての文に対して人手でクレーム／非クレームのラベル付けを行ったものを教師データとしてモデル学習に用いる。この手法で得られるデータでは核文と近接文の区別がないため、学習には通常のナイーブベイズ・モデルを用いる。提案手法と比べると、この手法では量は少量だが質の高い学習データが利用できる。このデータ作成作業は、評価実験の正解データ作成と同一の作業となる。ただし、このデータ作成には正解データの作成に従事した作業者のうちの 1 名によって執り行なった。作業時間は約 30 時間であった。

- 文書ラベルを教師データ作成に用いる手法（以下、文書ラベル）

本実験で使用しているレビューデータには、本研究でいうクレームとほぼ同等の概念を示している「苦情」というラベルがレビュー単位に付与されている。そこで、ここでは、文よりも粗い文書に対する教師情報を利用して、文単位の教師データを自動生成することを考える (Nigam and Hurst 2004)。具体的には、「苦情」ラベルが付与されたレビューに含まれている全ての文をクレーム文とみなし、逆に、「苦情」ラベルが付与されていないレビューに含まれている全ての文を非クレーム文とみなすことで教師データを自動生成し、モデル学習に用いる。モデルは先と同様の理由で通常のナイーブベイズ・モデルを用いる。提案手法と比べると、この手法では相対的に質は低いが、大量の学習データが利用できる。

- 辞書による手法

この手法は教師あり学習は行わず、辞書のエントリをルールとみなしたルールベース手法である。評価用データに対して 2.2 節で述べた核文ラベル付け、および 2.3 節で述べた近接文ラベル付けの手続きを直接適用してクレーム文を検出する。ただし、ここでの焦点はデータ生成時とは違って、クレーム文を検出できるか否かであるため、ラベル付けの結果、クレームとラベル付けされた文以外は全て非クレームであるとみなして評価し

た. なお, 辞書は 4.2 節で述べた辞書を用いる.

実験結果を図 5 に示す. また, 表 3 に提案手法のラベル付けと人手ラベル, 文書ラベルの各手法によるラベル付けの特徴をまとめる.

図 5 において, 辞書による手法は, ナイーブベイズモデルを用いた分類時に導入した閾値のパラメータが存在しないため, 11 点平均適合率を計算できない. そのため, ここでは再現率と適合率によって分類性能を評価している. なお, 図中の“提案ラベル”が提案手法の結果であり, さきほどの評価実験で最良であった拡張モデル NB+ctxBF(divide) で文脈長 $N = 2$ の実験結果を掲載している. また, “辞書 (核)” は, 辞書による手法のうち, 核文ラベル付けのみを考慮した場合の結果であり, “辞書 (核+近接)” が, 核文ラベル付けと近接文ラベル付けの両方を考慮した場合の結果である.

図 5 実験結果 (他手法との比較)

表 3 各ラベル付け手法で生成される教師データの特徴

ラベル付け	作成方法	教師データの質	教師データの量	実験で扱ったデータ数	
人手ラベル	手動	◎	×	2,000 文書	8,639 文
文書ラベル	自動	×	◎	347,000 文書	1,547,566 文
提案ラベル	自動	○	◎	347,000 文書	1,248,885 文

図 5 から、比較したどの手法よりも提案手法が良い性能を示していることがわかる。辞書による方法は、辞書に登録されている単語が含まれていない文に対しては適用できないため、再現率が低い。近接文を考慮することである程度の再現率を確保することは可能であるが、当然ながらその代償として適合率が下がる結果となっている。ここで、固有表現抽出課題がそうであるように、一般に、辞書に基づいた手法では再現率が低くなるがその一方で適合率が高くなる傾向がある。近接文の情報を用いない“辞書（核）”の結果は特にその傾向を示している。ただし、今回の実験結果では、再現率を固定させて適合率を見ると、ナイーブベイズ・モデルを用いた手法の方が適合率がより高い結果となっていた。これは、本研究課題では、辞書に登録されている一部の単語の情報だけでは文全体のクラス（クレーム／非クレーム）が正しく決定できない場合があり、このような場合には、辞書による方法よりも文内の単語情報を総合的に考慮できるナイーブベイズ・モデルが適していたためと考えられる。

次に、文書ラベルを利用する方法は、文書内のすべての文を教師データとして利用できる。そのため、提案手法と同程度かそれ以上の教師データが利用できるという特徴がある。しかし、文書内には一般的にクレームと非クレームが混在することから、文書ラベルと整合していない信頼性の低いデータを多く含む結果となり、そのことが性能の低下に繋がっていると考えられる。最後に、人手作成による方法は、もっとも質の高い教師データを準備することができるが、作成負荷の高さから、量を確保することが難しい。今回は人手で 2,000 件（8,639 文）のレビューから教師データを作成したが、提案手法を上回ることはなかった。

4.5 学習データとクレーム文検出精度の関係について

先でも述べたように、一般に、人手作成された教師データは質が高い反面、多くの量を準備することが困難である。一方、提案手法のように自動生成された教師データは人手作成されたデータよりも質が落ちるが、ラベルのない生データを準備するだけで手軽に増量できる。ここでは、人手によって教師データを作成する場合と第 2 節の提案手法によって教師データを自動生成する場合のそれぞれについて、教師データの量と分類性能の関係を調査する。なお、両方で教師データ以外の実験条件を合わせるために、この実験では、モデルには通常のナイーブベイズ・モデルを用いた。

実験結果を図 6 に示す。横軸が学習データ量（対数スケール）であり、縦軸が 11 点平均適合率である。どちらの実験結果についても、まず今回の実験において最大で利用可能なデータ量（人手ラベルの場合：レビュー 2,000 件、提案ラベルの場合：レビュー約 347,000 件）から性能測定を開始し、そこから一部の学習データを無作為に削除することで使用できる学習データ量がより少ない環境を設定して、これを繰り返しながらグラフをプロットした。

図 6 から、まず、どちらの手法においてもデータ量を増やすことで性能が向上することが確認できる。データ量が同じ場合は、当然のことながら、人手による方法の方が良い性能となる。

図 6 教師データ量の影響

しかし、提案手法によってデータ量を増加させることで、今回の場合は 10,000 件までデータ量を増やした時点で両者の性能が同等となり、さらにデータ量を増やすことで提案手法が人手による手法を上回ることができた。

この実験結果は、あくまでひとつのケース・スタディであり、具体的な数値自体に意味を求めることは困難であると考えられる。しかし、この結果は、人手によって十分な教師データが作成できない状況においては、自動生成手法を適用することで得られる教師データの量的利点という恩恵を受けられることを示唆していると言える。

5 関連研究

従来から、評判分析に関する研究を中心にして、意見を好評／不評に分類する研究が多くなされている (Pang and Lee 2008)。しかし、本論文では、応用面を重視した際、主に製品やサービスを提供する企業にとっては意見の好不評という側面だけでは十分でないことから、クレームという好不評とは異なる観点を導入し、意見を含むテキストからクレームという特定の意見を検出する手法について述べた。我々と同様に好評／不評以外の意見に着目した研究には、第 1 節で述べた永井らの研究 (永井 他 2002, 2003) の他にも幾つか存在する。例えば、金山 (金山, 那須川 2005) は、テキストから好評／不評の評判に加えて要望を抽出する手法を提案している。彼らの手法は、文に含まれる評判や要望を意図フレームと呼ばれる独自の形式に自動的

に変換しつつ抽出するようになっており、この変換・抽出処理において、既存の機械翻訳機構を再利用している。彼らの抽出対象である評判、要望の中に本研究におけるクレームも含まれていると考えられるが、彼らの手法は、機械翻訳機構が内部的に備える各種の言語知識のもとに成立しており、運用には人手による多大な管理負荷を要すると考えられる。一方で、本研究では極力人手の負荷を軽減することを指向しており、金山らの手法とはアプローチの方向性が異なる。また、他の関連研究として、自由記述アンケートから要求や要望を判定することに特化した大塚ら(大塚, 内山, 井佐原 2004)や山本ら(山本, 乾, 高村, 丸元, 大塚, 奥村 2006)の研究がある。彼らの論文に定義がないため厳密にはよくわからないが、彼らの扱っている要求や要望といった意見の分類クラスは、我々のクレームの一部分に該当すると考えられる。Goldberg ら (Goldberg, Fillmore, Andrzejewski, Xu, Gibson, and Zhu 2009) は、新年の願い事が集められたテキストコーパスから Wish (願望) を機械学習を用いて自動抽出する研究を行っている。

本研究では、レビュー中の各文をクレーム／非クレームに分類する課題に対して、ナイーブベイズ・モデルを採用し、データ特性に合わせて、その拡張を行った。拡張モデルでは、文間の周辺文脈をモデルに適切に反映させることができる。ここで、文書中の各文を対象とした分類問題を、文書中の文系列に対するラベリング問題とみなすことで、条件付確率場 (Conditional Random Fields; CRF) (Lafferty, McCallum, and Pereira 2001) のような、より高度なモデルを適用することについて検討する。まず、2 節で述べたデータ生成過程では、文書内のすべての文に対してラベルを付与するわけではなく、ある特定の文のみにラベルを付与することで教師データを作成する。そのため、CRF のような系列の構成要素についての全てのラベルを必要とするようなモデルは本研究の設定では直接は適用できない。坪井ら(坪井, 森, 鹿島, 小田, 松本 2009) によって、部分的なアノテーション情報から CRF の学習を行う手法が提案されており、この手法を適用することは不可能ではないが、彼らの手法を適切に適用するにあたり、アノテーションされている部分は人手による信頼性の高い情報であるという暗黙的な仮定が必要であると考えられ、データの自動生成を前提とする本研究の設定とは相性が良くないと考えられる。

6 おわりに

本論文では、レビュー文書からクレームが記述された文を自動検出する手法として、極力人手の負荷を軽減することを指向した次の 2 つの手法を提案した。(1) 評価表現と文脈一貫性に基づく教師データ自動生成手法。(2) 自動生成された教師データの特性を踏まえたナイーブベイズ・モデルの拡張手法。そして、評価実験を通して、これらの提案を組合せ、検出対象となる文の周辺文脈の情報を適切に捉えることで、クレーム文の検出精度を向上させることができ

ることを示した。また、人手によって十分な教師データが作成できない状況においては、提案したデータ自動生成手法を適用することで得られる教師データの量的恩恵を受けられることを示した。

本論文で議論ができなかった今後の課題としては、以下のような項目があげられる。

- 分類クラスの事前分布について：ナイーブベイズ・モデルでは、式 (2) にあるように、分類クラスの事前分布 $P(c)$ の情報を考慮する。しかし、本研究のように教師データを自動生成する際は事前分布 $P(c)$ はデータ自動生成手法に依存しており、本研究の場合では、利用する評価表現辞書の特徴に依存することになる。今後、評価表現辞書および事前分布 $P(c)$ と検出性能との関係について考察することが必要である。
- 各種のパラメータ調整について：本研究において、幾つかのパラメータは恣意的に指定している。例えば、考慮する周辺文脈の長さについて評価実験では可変させていたが、それらは、データ生成、モデル学習、分類の各過程で同期させている。しかし、原理的にはデータ生成時のみ文脈長を延長するといった設定も可能であり、これらの最適な調整は今後の課題である。
- 学習アルゴリズムについて：本研究では基本モデルとしてナイーブベイズ・モデルを採用して議論を進めたが、同様の議論を Support Vector Machine (SVM) (Vapnik 1995) のような別の学習アルゴリズムを用いて行うことも興味深い。ただし、SVM にはモデルの学習速度が遅いという欠点がある。そのため、提案手法の利点である大規模な教師データを自動生成できるという点を活かすためには SVM の高速学習を含めた総合的な検討が必要である。
- クレームの内容分類について：本研究はクレーム検出をクレームであるか否かという 2 値分類問題として扱った。しかし、実利用環境で検出されたクレームを企業内で活かしていくには、クレーム内容も合わせて自動分類できることが望ましい。例えば、対象が宿泊施設の場合では、「部屋」や「食事」といったクレームの対象に関する分類クラスを別途設定し、これらも同時に考慮した検出モデルを検討することも興味深い。
- 見逃し状況について：クレームを見逃す状況として、本論文では、レビュー文書内に部分的に現れるクレームの見逃しについて扱った。しかし、第 1 節でも述べたように、多対一型のコミュニケーションに起因する見逃しへの対処も重要である。今後、多対一型のコミュニケーションに起因する見逃しに対する提案手法の適用可能性についても検討したい。

謝 辞

本研究を遂行するにあたり，楽天株式会社楽天技術研究所の新里圭司氏，平手勇宇氏，山田薫氏から示唆に富む多くの助言を頂きました。諸氏に深く感謝いたします。また，実験にあたり，楽天トラベル株式会社から施設レビューデータを提供して頂きました。ここに記して感謝の意を表します。

参考文献

- Goldberg, A. B., Fillmore, N., Andrzejewski, D., Xu, Z., Gibson, B., and Zhu, X. (2009). “May All Your Wishes Come True: A Study of Wishes and How to Recognize Them.” In *Proceedings of the Human Language Technology Conference and the North American Chapter of the Association for Computational Linguistics*, pp. 263–271.
- 乾孝司，奥村学 (2006). テキストを対象とした評価情報の分析に関する研究動向. 自然言語処理, **13** (3), pp. 201–241.
- 金山博，那須川哲哉 (2005). 要望表現の抽出と整理. 言語処理学会第 11 回年次大会発表論文集, pp. 660–663.
- Lafferty, J., McCallum, A., and Pereira, F. (2001). “Conditional random fields: Probabilistic models for segmenting and labeling sequence data.” In *Proceedings of the 29th International Conference on Machine Learning*, pp. 282–289.
- Manning, C. D. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. The MIT Press.
- Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.
- 永井明人，高山泰博，鈴木克志 (2002). 単語共起照合に基づくクレーム抽出方式の改良. 情報科学技術フォーラム, pp. 113–114.
- 永井明人，増塩智宏，高山泰博，鈴木克志 (2003). インターネット情報監視システムの試作. 情報処理学会研究報告. 自然言語処理研究会報告, **2003** (23), pp. 125–130.
- 那須川哲哉，金山博 (2004). 文脈一貫性を利用した極性付評価表現の語彙獲得. 情報処理学会研究報告. 自然言語処理研究会報告, **2004** (73), pp. 109–116.
- Nigam, K. and Hurst, M. (2004). “Towards a Robust Metric of Opinion.” In *AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications*, pp. 98–105.
- 大塚裕子，内山将夫，井佐原均 (2004). 自由回答アンケートにおける要求意図判定基準. 自然言語処理, **11** (2), pp. 21–66.

Pang, B. and Lee, L. (2008). *Opinion Mining and Sentiment Analysis - Foundations and Trends in Information Retrieval Vol.2, Issue 1-2*. Now Publishers Inc.

高村大也, 乾孝司, 奥村学 (2006). 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, **47** (2), pp. 627–637.

坪井祐太, 森信介, 鹿島久嗣, 小田裕樹, 松本裕治 (2009). 日本語単語分割の分野適応のための部分的アノテーションを用いた条件付確率場の学習. 情報処理学会論文誌, **50** (6), pp. 1622–1635.

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer.

山本瑞樹, 乾孝司, 高村大也, 丸元聡子, 大塚裕子, 奥村学 (2006). 文章構造を考慮した自由回答意見からの要望抽出. 言語処理学会第 12 回年次大会併設ワークショップ「感情・評価・態度と言語」.

略歴

乾 孝司：2004 年奈良先端科学技術大学院大学情報科学研究科博士課程修了。日本学術振興会特別研究員，東京工業大学統合研究院特任助教等を経て，2009 年筑波大学大学院システム情報工学研究科助教。現在に至る。博士（工学）。近年は CGM テキストに対する評判分析に興味をもつ。

梅澤 佑介：2011 年筑波大学情報学群情報メディア創成学類卒業。2013 年筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻修了。同年 4 月から株式会社ヤフー。在学中は自然言語処理の研究に従事。

山本 幹雄：1986 年豊橋技術科学大学大学院修士課程修了。同年株式会社沖テックノシステムズラボラトリ研究開発員。1988 年豊橋技術科学大学情報工学系教務職員。1991 年同助手。1995 年筑波大学電子・情報工学系講師。1998 年同助教授。2008 年筑波大学大学院システム情報工学研究科教授。博士（工学）。自然言語処理の研究に従事。言語処理学会，人工知能学会，ACL 各会員。

(2013 年 5 月 9 日 受付)

(2013 年 8 月 2 日 再受付)

(2013 年 9 月 17 日 採録)