複数文書からの文レベルの情報の書き漏らしの検出

岡﨑 健介 [*1]　村田 真樹 [*1,2]　馬 青 [*3]
[*1] 鳥取大学大学院 工学研究科 情報エレクトロニクス専攻
[*2] 鳥取大学工学部付属クロス情報科学研究センター
[*3] 龍谷大学 理工学部 数理情報学科
[*1]{s142017,murata}@ike.tottori-u.ac.jp
[*3] qma@math.ryukoku.ac.jp

1　はじめに

文書に書かれるべき情報を書き漏らしていることを指摘することは文書作成を行う際の支援として有効であると考えられる．岡田らの研究 [1] では，論文に記載すべき情報の欠落をルールベースによる手法と機械学習による手法を用いて自動で検出・修正することで論文の文章作成支援を行った．赤野らの研究 [2] では，単語の情報をクラスタリングを用いて表に整理し，表の空欄箇所を情報の書き漏らし箇所として検出した．野浪らの研究 [3] では，赤野らの研究と同様にして単語の情報を表に整理し，表の空欄箇所を情報の書き漏らし箇所として検出した後，表の空欄箇所に該当する情報を Web ページから抽出していた．赤野や野波らの研究が単語の情報を対象としているのに対し，過去の我々の研究 [4] では，複数の文書に含まれる文の情報をクラスタリングを用いて表に整理した．

本研究では，図 1 のように，過去の研究 [4] の手法を用いて，関連する文書に含まれる文の情報を共通する情報ごとに表に整理することで，情報の書き漏らしを検出・指摘し，これによって文書作成を支援する．

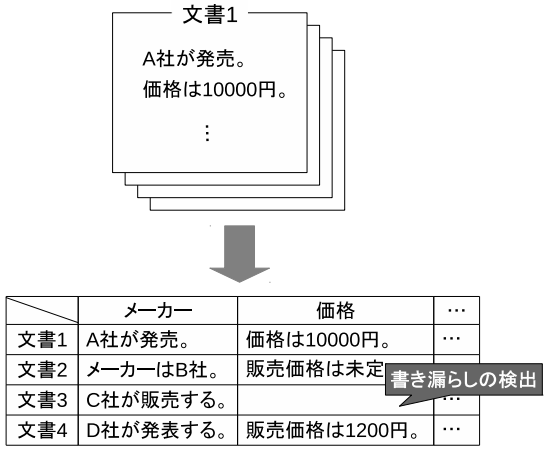

図 1　書き漏らしの例

2　書き漏らしの指摘

文書中の書き漏らしの検出を行うために，まず，関連する内容の複数の文書に含まれる文の情報を過去の研究 [4] で提案した方法を用いて共通する情報ごとに表に整理する．そして表の空欄となっている箇所を検出し，これを指摘することにより文書作成の支援を行う．

2.1　表の生成手順

複数の文書に含まれる文の情報を表に整理する手順を以下に示す．また，手順の概要図を図 2 に示す．

手順 1　複数文書を文単位に分割する．
手順 2　手順 1 で分割した文のベクトルを計算する．
手順 3　文ベクトルを x-means 法 [5, 6] でクラスタリングする．
手順 4　クラスタの重要度を計算し，重要度の高い順に，行を文書，列をクラスタとする表に整理することで可読性を高める．
手順 5　表の各クラスタに項目名を付与し，クラスタにどのような情報が含まれるかをわかるようにする．

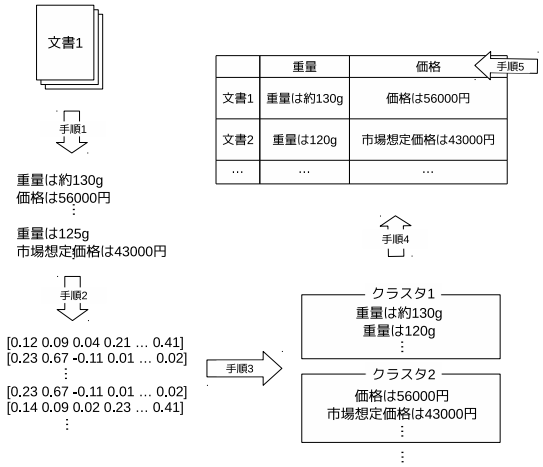

図 2　手順の概要図

2.2 文の分割方法

2.1 節の手順 1 における複数文書を文単位に分割する方法を説明する．文書を句点ごとに分割したものを文とした場合，例えば「人口は 98,891 人で，面積は 611.76km^2。」のような複数の情報を含む文が存在してしまう．このような文は，「人口は 98,891 人。」という人口に関する文と，「面積は 611.76km^2。」という面積に関する文に分割されることが望ましい．よって以下の手順で文を分割し，得られた短い文を本研究では 1 つの文として扱う．図 3 に分割結果の例を示す．

1. 文を KNP[*1]を用いて構文解析する．
2. 条件 (a)，(b) を同時に満たす文節箇所で分割する．
   （a）文節の係り先が末尾の文節番号である．
   （b）並列構造を表す<P>が付与されている．
3. 分割された文に対しても，文を分割できなくなるまで 1，2 を行う．
4. 分割された各文を KNP で格解析する．
5. 出力された格解析結果のうち，係り先が末尾の文節番号である文節，もしくは末尾の文節に注目する．
6. 注目している格解析結果に含まれる各格要素について，述語よりも前にある場合は，格要素を格要素に係る文節と統合する．
7. 格要素と述語をまとめて文を作る．

分割前

流域には貴重な生態系が拡がっていたが、噴火によって大半の渓谷が分厚い火山堆積物の底に埋もれた。

分割後

貴重な生態系が流域に拡がっていた。
噴火により大半の渓谷が分厚い火山堆積物の底に埋もれた。

図 3　分割結果の例

2.3 文ベクトルの計算

2.1 節の手順 2 における文ベクトルの計算方法を説明する．文ベクトルは以下の手順で求める．

1. 文を MeCab[*2]を用いて形態素解析する．
2. 形態素解析結果のうち，品詞が名詞で，かつ，品詞分類 1 が代名詞，数，非自立，副詞可能でない単語を抽出する．
3. 抽出した単語のベクトルの総和を文ベクトルとする．(単語ベクトルの計算は 2.3 節で述べる．)

2.4 単語ベクトルモデル

2.3 節の文ベクトルの計算で用いる単語のベクトルには，fastText[7] を用いた．

今回は学習データとして，アルファベットとカタカナは全角に，英数字は半角に統一した Wikipedia の全記事データ (2017 年 6 月 1 日時点) を使用した．データの詳細を表に示す．また，単語ベクトルの次元数は 300 次元とした．

2.5 x-means 法

本研究では文ベクトルのクラスタリングに x-means 法を用いる．x-means 法は，k-means 法を拡張した手法である．k-means 法では，あらかじめクラスタの数を指定する必要があるが，x-means 法では以下の手順により，最適なクラスタの数を推測できる．

1. クラスタ数 $k = 2$ で再帰的に k-means 法を実行する．
2. クラスタリング前後のベイズ情報量基準 BIC を比較する．
3. BIC の値が小さくなる限りこれを続ける．

2.6 重要度の計算方法

2.1 節の手順 4 におけるクラスタごとの重要度の計算方法を説明する．

クラスタリング結果には関連する文だけで構成される密集率の高いクラスタもあれば，関連性のない文で構成される密集率の低いクラスタもある．密集率の高いクラスタほど重要であると考えられる．よって，k 番目のクラスタの密集率 d_k を式 (1) のように定める．ここで，N_k は k 番目のクラスタに含まれる文の総数であり，$S_{k,l}$ は k 番目のクラスタに含まれる l 番目の文のベクトルであり，$S_{k,mean}$ は k 番目のクラスタに含まれる文のベクトルの平均である．

$$d_k = \frac{1}{N_k} \sum_{l=1}^{N} \frac{S_{k,l} \cdot S_{k,mean}}{|S_{k,l}||S_{k,mean}|} \tag{1}$$

多くの文書の情報を含むクラスタほど重要であると考えられる．よって，k 番目の文書カバー率 c_k を式 (2) のように定める．p_k は k 番目のクラスタにおいて文を抽出できた文書の数であり，P は文書の総数である．

$$c_k = \frac{p_k}{P} \tag{2}$$

k 番目のクラスタの重要度 i_k を式 (3) のように定義

[*1] http://nlp.ist.i.kyoto-u.ac.jp/index.php?KNP
[*2] http://taku910.github.io/mecab/

する．

$$i_k = d_k \times c_k \tag{3}$$

2.7 クラスタの項目名の求め方

生成された表の各クラスタについて，以下の手順でクラスタの項目名を付与する．

1. クラスタに含まれるの各文について，文に含まれる単語のうち品詞が名詞のものを抽出する．
2. 1 で抽出した各単語について，文書頻度を求める．
3. 文書頻度が最大の単語をクラスタの項目名として付与する．
4. 文書頻度が最大の単語が複数ある場合は，読点で区切って全て付与する．

3 実験

関連する内容の複数文書から 2.1 節の手順で情報を抽出し表を生成する．なお，x-means 法の初期値依存性により，生成される表はプログラムの実行ごとに異なるため，表を 5 つ生成し，その中から人手で最適なものを選択した．選択した表のうち空欄となっている箇所を情報の書き漏らし箇所として検出する．検出された空欄箇所について，対応する文書中に該当する情報がない場合，正しく書き漏らしを指摘できたと判断する．

3.1 実験データ

実験で用いる複数文書には，価格.com のスマートフォンの新製品記事 20 件と，事件に関する毎日新聞の強盗事件に関する新聞記事 20 件の 2 種類を使用した．各データの詳細を表 1 に示す．

表 1　複数文書の詳細

	文書数	文数	1 文の平均文字数
新製品記事	20	313	46.35
新聞記事	20	128	39.25

3.2 実験結果

新製品記事を複数文書として用いた場合に生成された表の一部を表 2 に示す．新製品記事での結果として 28 列の表が得られた．また，新聞記事では 10 列の表が得られた．

表 2　生成された表の一部 (新製品記事)

	重量	ＯＳ
文書 1	重量は約１４３ｇ	ＯＳは「Ａｎｄｒｏｉｄ７．０」をプリインストールする
文書 2	重量は１５８ｇ	
文書 3	重量は約１９８ｇ	ＯＳは「Ａｎｄｒｏｉｄ７．１」をプリインストールした
文書 4	重量は約１５０ｇ	ＯＳは「Ａｎｄｒｏｉｄ７．１」をプリインストールする

3.3 情報抽出の性能の評価

実験で得られた 2 種類の表それぞれの重要度の高い上位 5 列に対して，情報抽出の性能を適合率と再現率，F 値を用いて評価する．適合率は式 (4)，再現率は式 (5) を用いて算出する．F 値は適合率と再現率の調和平均である．評価結果を表 3，表 4 に示す．

$$\text{適合率} = \frac{\text{列の項目名に関連する文の数}}{\text{列に含まれる文の数}} \tag{4}$$

$$\text{再現率} = \frac{\text{列の項目名に関連する文の数}}{\text{列の項目名に関連する文書中の文の数}} \tag{5}$$

表 3　新製品記事での評価結果

列番号	列 1(重量)	列 2(OS)	列 3(メモリ)	列 4(容量)	5(画素)	平均
適合率	1.00(15/15)	1.00(13/13)	0.54(13/24)	1.00(15/15)	0.85(17/20)	0.88
再現率	0.88(15/17)	0.81(13/16)	0.76(13/17)	0.83(15/18)	0.55(17/31)	0.77
F 値	0.94	0.90	0.63	0.91	0.67	0.81

表 4　新聞記事での評価結果

列番号	列 1(捜査)	列 2(男)	列 3(身長)	列 4(男性, けが)	列 5(姿)	平均
適合率	0.95(20/21)	0.86(19/22)	1.00(13/13)	1.00(9/9)	0.80(8/10)	0.92
再現率	1.00(20/20)	0.69(19/28)	1.00(13/13)	0.45(9/20)	0.62(8/13)	0.75
F 値	0.98	0.76	1.00	0.62	0.70	0.81

3.4 書き漏らし箇所の検出精度の評価

実験で得られた 2 種類の表の空欄を 1 つ以上含む列のうち重要度の高い上位 5 列に対して，正しく書き漏らし箇所を検出できたかを適合率と再現率，F 値を用いて評価する．適合率は式 (6)，再現率は式 (7) を用いて算出する．F 値は適合率と再現率の調和平均である．F 値が大きいほど書き漏らし箇所を正しく検出できたことを意味する．例えば，文書 3 に重量に関する情報がない場合，作成された表が表 5 であれば，文書 3 の欄が空欄となっているので「項目名に関連する文を含まない文書の欄が空欄である」といえる．一方，表 6 のような表が作成された場合は，文書 3 の欄に重量とは関連しない情報が含まれており，「項目名に関連する文を含まない文書の欄が空欄である」とはいえない．表 7 の場合は，重量に関する情報を含む文書 2 の欄が空欄となっており「列に含まれる空欄の数」が大きくなり，適合率が低い値となる．各表の重要度の高い上位 5 列に対する評価結果を表 8，表 9 に示す．ベースラインとして，各表の各列を全て空欄とした場合の評価結果を表 10，表 11 に示す．

$$\text{適合率} = \frac{\text{項目名に関連する文を含まない文書の欄が空欄である数}}{\text{列に含まれる空欄の数}} \tag{6}$$

$$\text{再現率} = \frac{\text{項目名に関連する文を含まない文書の欄が空欄である数}}{\text{列の項目名に関連する文を含まない文書の数}} \tag{7}$$

表 5　正しく書き漏らしを検出した例

	重量
文書 1	重量は約 130g
文書 2	重量は 125g
文書 3	
文書 4	重量は 138g

表 6　書き漏らしの検出を誤った例 1

	重量
文書 1	重量は約 130g
文書 2	重量は 125g
文書 3	メモリーが 3GB
文書 4	重量は 138g

表 7　書き漏らしの検出を誤った例 2

	重量
文書 1	重量は約 130g
文書 2	
文書 3	
文書 4	重量は 138g

表 8　新製品記事での評価結果

列番号	列 1(重量)	列 2(OS)	列 3(メモリ)	列 4(容量)	列 5(画素)	平均
適合率	0.80(4/5)	0.67(4/6)	0.67(4/6)	0.60(3/5)	0.40(2/5)	0.63
再現率	1.00(4/4)	1.00(4/4)	1.00(4/4)	1.00(3/3)	1.00(2/2)	1.00
F 値	0.89	0.80	0.80	0.75	0.57	0.76

表 9　新聞記事での評価結果

列番号	列 3(身長)	列 4(男性, けが)	列 5(姿)	列 6(売上金)	列 7(運転手)	平均
適合率	1.00(7/7)	0.08(1/12)	0.73(8/11)	0.25(3/12)	0.08(1/13)	0.43
再現率	1.00(7/7)	1.00(1/1)	1.00(8/8)	1.00(3/3)	1.00(1/1)	1.00
F 値	1.00	0.15	0.84	0.40	0.14	0.51

表 10　新製品記事での評価結果 (ベースライン)

列番号	列 1(重量)	列 2(OS)	列 3(メモリ)	列 4(容量)	列 5(画素)	平均
適合率	0.20(4/20)	0.20(4/20)	0.20(4/20)	0.15(3/20)	0.10(2/20)	0.17
再現率	1.00(4/4)	1.00(4/4)	1.00(4/4)	1.00(3/3)	1.00(2/2)	1.00
F 値	0.33	0.33	0.33	0.26	0.18	0.29

表 11　新聞記事での評価結果 (ベースライン)

列番号	列 3(身長)	列 4(男性, けが)	列 5(姿)	列 6(売上金)	列 7(運転手)	平均
適合率	0.35(7/20)	0.05(1/20)	0.40(8/20)	0.15(3/20)	0.05(1/20)	0.20
再現率	1.00(7/7)	1.00(1/1)	1.00(8/8)	1.00(3/3)	1.00(1/1)	1.00
F 値	0.52	0.10	0.57	0.26	0.10	0.31

3.5　考察

3.4 節から，本研究の手法に対する評価結果は，新製品記事を入力とした場合の F 値が 0.76 で，新聞記事の場合の F 値は 0.51 であった．また，ベースラインでの結果は，新製品記事を入力とした場合の F 値が 0.29 で，新聞記事の場合の F 値は 0.31 であった．いずれの文書の場合も，ベースラインよりも本研究の手法のほうが高い精度で情報の書き漏らしを検出することができた．一方で，いずれの文書の場合も，再現率の平均が 1.00 であるのに比べ，適合率がそれぞれ 0.63 と 0.43 であり低い傾向にあった．この原因としては，表生成の際に行うクラスタリングにおいて，内容が関連する文が異なるクラスタに割り当てられたため，表 12 のように列に多くの空欄が生じたことが原因であると考えられる．このような問題は関連する文同士のベクトルの違いが大きいために生じると思われる．例えば「男性店員は重症」と「運転手にけがはなかった」という文のように，「けが」という共通の内容を表す文同士であっても含まれる単語が異なる場合に，これらの文のベクトルの違いが大きくなり，クラスタリングを行っても同じクラスタに割り当てられない場合が生じる．このような問題を解消するためには，文のベクトルを計算する際に用いる文中の単語の重要度を考慮する必要があると考えられる．文中の単語の中には文の持つ情報をよく表す単語もあればそうでない単語もある．文の持つ情報をよく表す単語ほど重みを大きくしたうえで文ベクトルを計算することで，文の情報をより表した文ベクトルが得られると考えられる．

表 12　関連する文が異なるクラスタに割り当てられた例

	列 4(男性, けが)	列 7(運転手)
文書 1	店員にけがはなかった	
文書 2	男性店員は重傷	
文書 3		運転手にけがはなかった
文書 4		運転手にけがはなかった

4　おわりに

本研究では，関連する文書に含まれる文の情報を共通する情報ごとに表に整理することにより，情報の書き漏らしの検出を行った．本研究の手法に対する評価結果は，新製品記事を入力とした場合の F 値が 0.76 で，新聞記事の場合の F 値は 0.51 であった．また，ベースラインでの結果は，新製品記事を入力とした場合の F 値が 0.29 で，新聞記事の場合の F 値は 0.31 であり，ベースラインよりも本研究の手法のほうが高い精度で情報の書き漏らしを検出することができた．一方で，いずれの文書を用いた場合も，再現率に比べ適合率が低い値にとどまっているため，文のベクトルを改善し，クラスタリングの際に関連する文同士が異なるクラスタに割り当てられることを解消する必要がある．

謝辞

本研究は科研費 (26330252) の助成を受けたものである．

参考文献

[1] 岡田拓真, 村田真樹, 馬青. 論文における記載不備の自動修正に向けた分析. 言語処理学会第 23 回年次大会, pp. 573–576.
[2] Akano Hokuto, Murata Masaki, and Ma Qing. Detection of inadequate descriptions in wikipedia using information extraction based on word clustering. IFSA-SCIS 2017, pp. 1–6, 2017.
[3] 野浪尚哉, 村田真樹, 馬青. 検索エンジンを用いた記載欠落箇所の補完. 言語処理学会第 23 回年次大会, pp. 971–974.
[4] 岡崎健介, 村田真樹, 馬青. 複数文書からの重要情報の抽出と表の作成. 言語処理学会第 24 回年次大会, pp. 240–243.
[5] Dan Pelleg and Andrew Moore. X-means: Extending k-means with efficient estimation of the number of clusters. In *ICML '00 Proceedings of the Seventeenth International Conference on Machine Learning*, pp. 727–734, 2000.
[6] 石岡恒憲. x-means 法改良の一提案 : k-means 法の逐次繰り返しとクラスターの再併合. 計算機統計学, 第 18 巻, pp. 3–13, 2006.
[7] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. In *Transactions of the Association for Computational Linguistics*, Vol. 5, pp. 135–146, 2017.