抽出型文書要約における分散表現の学習 ―文書と要約の距離最小化―

田口 雄哉　　重藤 優太郎　　新保 仁　　松本 裕治
奈良先端科学技術大学院大学
{taguchi.yuya.to0, yutaro-s, shimbo, matsu}@is.naist.jp

1 はじめに

1.1 背景

自然言語処理における文書要約は単一もしくは複数の文書が与えられ，その文書を長さに関する制限 (100 単語以下など) のもとで要約するタスクである．文書要約には，大きく分けて生成型要約と抽出型要約の 2 種類の方法がある．生成型要約は，入力文書には存在しない単語や構文を用いて要約を生成し，抽出型要約は，入力文書に含まれる，その文書の内容を十分に説明している代表的な文 (もしくは単語) を抽出する [12]．

近年，word2vec [10] などに代表される単語分散表現を用いて抽出型要約を行なう手法が提案されている [11, 4, 5]．文書要約においては，要約に含める文は要約内の他の文と意味的に重複していない方が良い．そのため，文を選択する際に文間の類似度をどのように測るかが課題となる．

bag-of-words ベクトルで文を表現した場合は，単語の表層が一致していなければ文間の類似度が測れない．一方，単語分散表現を用いることで表層上異なっていても文間の類似度を測ることが可能になる [4]．

多くの自然言語処理のタスクにおいて，単語分散表現をタスクの目的関数に応じて学習することで性能の向上が報告されている [9, 6]．抽出型文書要約においても同様に，文書要約に適した単語分散表現を獲得することで，要約性能の改善が期待できる．

1.2 研究目的と貢献

本研究の目的は，文書要約タスクに適した単語分散表現を学習を行い，それによる精度の向上が見られるかを検証することである．

具体的には，単語の分散表現を用いて文書と参照要約をベクトル空間に埋め込み，その空間において，文書ベクトルと参照要約ベクトル間の距離の最小化を行う．この距離最小化の操作は，実際には，単語の分散表現学習に帰着するので，結果として，文書要約に適した単語分散表現を獲得することができると思われる．

5 節で示すが，既存の抽出型要約手法を用いて，提案手法により単語分散表現を学習した場合と学習しない場合を比べた結果，学習した場合のほうが良い要約精度を得ることが確認できた．

2 関連研究

以下では，本研究と関連する単語分散表現に基づく抽出型要約の研究を説明する．

Kågebäck ら [4] は，Lin ら [8] が定義した評価関数を基に，文間の類似度を単純な bag-of-words ではなく，単語分散表現のコサイン類似度に基づく評価関数を提案した．

Kobayashi ら [5] も，単語分散表現を用いたコサイン類似度を提案しているが，Kågebäck らとは異なり，文書と抽出された文集合の間に類似度を定義し，その類似度をそのまま評価値として使用している．本実験では，Kobayashi らの抽出手法を用いるため，この手法の詳細を 3 節で詳しく説明する．

Kågebäck らと Kobayashi らの研究では，文選択時の類似度尺度を開発することに注目しており，単語分散表現の学習は行っていない．この点で，分散表現の学習を行なう本研究とは異なる．

本研究と同じように単語分散表現の学習を行っている研究がある [14, 1]．それらは各文の ROUGE 値を予測する回帰問題として定式化している．そのような回帰モデルによる手法は，事前に文書内の各文に対して ROUGE 値を求める必要がある．それに対し，本研究では文書内の各文に対して ROUGE 値を計算せずに要約のタスクに合った分散表現の学習法を提案する．

Algorithm 1 抽出アルゴリズム

Input: Document $\mathcal{D}$, length limit L,
and scaling paramter r
Output: Summary $\mathcal{C}$

$\mathcal{U} \leftarrow \mathcal{D}$
while $\mathcal{U} \neq \emptyset$ **do**
  $\mathcal{S}^* = \arg\max_{\mathcal{S} \in \mathcal{U}} \frac{\cos(\mathcal{S} \cup \mathcal{C}, \mathcal{D}) - \cos(\mathcal{C}, \mathcal{D})}{|\mathcal{S}|^r}$
  if $\sum_{\mathcal{S} \in \mathcal{C}} |\mathcal{S}| + |\mathcal{S}^*| \leq L$ **then**
    $\mathcal{C} \leftarrow \mathcal{C} \cup \mathcal{S}^*$
    $\mathcal{U} \leftarrow \mathcal{U} \setminus \mathcal{S}^*$
  else
    break
$\mathcal{S}^* \leftarrow \arg\max_{\mathcal{S} \in \mathcal{D}} \cos(\mathcal{S}, \mathcal{D})$
$\mathcal{C} \leftarrow \arg\max_{\mathcal{T} \in \{\mathcal{C}, \{\mathcal{S}^*\}\}} \cos(\mathcal{T}, \mathcal{D})$
return C

3 単語分散表現に基づく抽出型要約

まず，はじめにいくつか記号を定義する．単語を $w \in \mathcal{V}$ とし，文に含まれている単語の集合 $\mathcal{S} = \{w_1, \ldots, w_n\}$，文書を文集号 $\mathcal{D} = \{\mathcal{S}_1, \ldots, \mathcal{S}_m\}$ で表す[1]．また，単語 w の分散表現を $\mathbf{w} \in \mathbb{R}^d$ とみなす．

この表記を用いると，単一文書を対象とした抽出型要約は，入力として文書 $\mathcal{D} = \{\mathcal{S}_1, \ldots, \mathcal{S}_m\}$ が与えられ，そこから要約となる文の集合 $\mathcal{C} = \{\mathcal{S}_i, \ldots\}$ を抽出するタスクとして定式化される[2]．

Kobayashi らは，文書 $\mathcal{D}$ のベクトルを，その文書を構成する文に含まれる単語 w の分散表現 $\mathbf{w}$ を用いて以下のように定義した:

$$f(\mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{\mathcal{S} \in \mathcal{D}} \frac{1}{|\mathcal{S}|} \sum_{w \in \mathcal{S}} \text{tf-idf}(w) \times \mathbf{w}. \quad (1)$$

ここで tf-idf(w) は単語 w の td-idf 値を返す関数を表している．この式は，文書だけでなく，要約となる文集合 $\mathcal{C}$ に対しても同様に適応することができる．

Kobayashi らは，文を抽出するか否かを判断するために，式 1 を用いて文書および要約となる文集合のコサイン類似度を以下のように定義し:

$$\cos(\mathcal{C}, \mathcal{D}) = \frac{\langle f(\mathcal{C}), f(\mathcal{D}) \rangle}{\| f(\mathcal{C}) \| \| f(\mathcal{D}) \|},$$

この類似度を，評価値として利用した．

具体的な文抽出の手順をアルゴリズム 1 に示す．

4 提案手法

1 節冒頭で述べたが，抽出型文書要約とは，与えられた文書に対し，長さの制限の中で最も文書全体の内容を言い表しているような文の部分集合を求めるタスクなので，(i) 文書とその要約が与えられ，かつ，それらがベクトル空間に埋め込まれている場合，それらの距離は小さい方が望ましい．

[1] n, m は文，文書によって異なる．
[2] 説明の簡略化のため，以後単一文書要約を仮定する

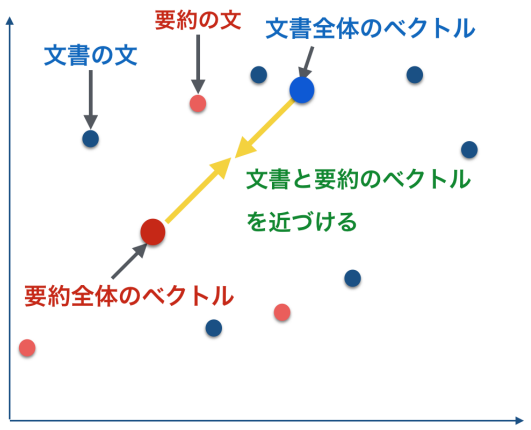

図 1: 分散表現を用いて文書および要約に含まれる文，さらに文書および要約全体をベクトル化し，2 次元にプロットし，距離の最小化を行なうイメージ図．

また，抽出型要約は，入力文書に含まれる重要な文を抽出するタスクなので (i) を仮定することができる時，(ii) 抽出すべき文，すなわち要約となり得る文とは，文書ベクトルと最も距離が近い文だと言える．

本節では，この考えを単語の分散表現に基いて実現する方法を述べる．図 1 は文書および要約の文ベクトル，そして文書ベクトルと要約ベクトルを用いた距離最小化のイメージである．

まず，データセットとして $\mathcal{P} = \{(\mathcal{D}_i, \mathcal{R}_i)\}_{i=1}^N$ が与えられることを考える．この $\mathcal{R}_i$ は文書 $\mathcal{D}_i$ の参照要約 (人手で作られた要約文の集合) を表す．

次に，要件 (i) を満たすべく，文書と要約の距離最小化を考える．文書 $\mathcal{D}$ および $\mathcal{C}$ が，式 (1) によって，ベクトル空間に埋め込まれている時，その距離の最小化は，以下の単語の分散表現に関する最適化問題となる (単語ベクトルを並べた行列を $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_{|\mathcal{V}|}] \in \mathbb{R}^{d \times |\mathcal{V}|}$ とする):

$$\min_{\mathbf{W}} \sum_{(\mathcal{D}, \mathcal{R}) \in \mathcal{P}} \| f(\mathcal{D}) - f(\mathcal{R}) \|_F^2 + \lambda \| \mathbf{W} - \mathbf{W}_0 \|_F^2. \quad (2)$$

第二項目は，$\mathbf{W} = \mathbf{0}$ が解となることを防ぐための正則化項である．この正則化は，$\mathbf{W}_0$ を word2vec などの学習済みの単語ベクトルにすることで，与えられた学習済み単語ベクトルからどの程度の変化を許容するかを調整する．これは，word2vec などの訓練済みの分散表現が，ある程度良いベクトル表現となっていることを仮定しているからである．

この式 (2) は，凸関数であり，また，最適な $\mathbf{W}$ は閉形式で導出することができる．

この学習により，要件 (i) が満たされていることを期待した上で，評価時には，獲得した $\mathbf{W}$ を用い，要約の抽出を行う．実際には 3 節で説明したアルゴリズムを用いる．これは，Kobayashi らの手法は，最も文書と類似した文を要約として抽出するので，要件 (ii) を満たすことになるからである．

5 実験

本実験では提案した単語分散表現の学習法が有効かどうかを検証するために，一般的な文書要約のベースライン手法に加え，既存の単語分散表現を用いた抽出型要約の手法と，単語分散表現の学習を行なったものをそれぞれ用いて比較を行なう．

5.1 実験設定

本研究では，単語分散表現を用いた抽出型要約の既存研究 [5, 4, 11] が使用している Opinosis データセット [3] を用いて 100 単語以内の要約を構成する．

このデータセットは全部で 51 のトピックがあり，それぞれのトピックごとにホテルや電化製品などに関するユーザーレビューが 50〜575 文あり，各トピックに人手て 4〜5 つの参照要約が付与されている．

Opinosis データセット自体は訓練データ，テストデータなどの分割が明示的に行われていない．そこで本研究では，全 51 トピック (文書) を 6:2:2 の比率で訓練データ，開発データ，評価データに分割し，5 回の交差検定を行って精度を検証する．

実験で用いる単語分散表現には，公開されている 300 次元の word2vec[3] と 50 次元の SENNA[4] を用いた．

データセットの前処理として，まず文字を全て小文字に変換し，その後，NLTK [5] を用いてストップワードの除去および単語分割を行なった．

要約の評価は ROUGE [7] で行なった[6]．ROUGE-N は 1〜4 を計測し，さらに skip-bigram を考慮する ROUGE-SU4 の計 5 尺度で評価を行なった．

5.2 比較手法

本実験では以下で説明する 5 手法を用いて評価を行なう．

LexRank Erkan ら [2] によって提案されたグラフベースの要約手法で，文書要約のベースラインとして一般的な手法である．文を TF-IDF ベクトルで表現し，文同士をコサイン類似度を基にエッジをつなぎ，Pagerank[13] を計算する．その後，高い固有値を持つ文から逐次要約に追加していく．

TF-IDF Lin ら [8] の劣モジュラ関数最大化による要約手法．彼らの手法は文を要約に追加するか判断する際に，文間の類似度を測るのに TF-IDF ベクトルを用いたコサイン類似度を使用している．

Embedding Kågebäck ら [4] の手法．Lin ら [8] の劣モジュラ関数を用い，TF-IDF ベクトルではなく単語分散表現を用いてコサイン類似度を計算している．Kågebäck らの手法は公開されている実装[7]を使い，単語分散表現には word2vec を用いた．

Cosine (W2V), Cosine (SENNA) 3 節で説明した，Kobayashi ら [5] の手法．単語分散表現には，word2vec および SENNA をそれぞれ用いた．

Cosine (W2V-Opt), Cosine (SENNA-Opt) 本論文で提案している手法．文書要約における文選択のアルゴリズムについてはアルゴリズム 1 を用いる．既存手法との違いはコサイン類似度を求める際に，学習された単語分散表現を使用している点である．

提案手法である単語分散表現の学習を行なう際には，word2vec と SENNA を $\mathbf{W}$ の初期値，および $\mathbf{W}_0$ とした．学習する単語分散表現 $\mathbf{W}$ は訓練済み単語分散表現内に登録されている単語のみを用い，それ以外の単語は未知語 (unk) として扱った．

アルゴリズム 1 における r と λ (式 (2)) はハイパーパラメータであり，開発データで ROUGE-2 が最も高いものを使用した．

5.3 実験結果

実験結果を表 1 に示す．

表 1 より，ROUGE-3, 4 を除く指標ではベースラインと比較して，word2vec の分散表現を基に学習したを手法 (W2V-Opt) が最も高いスコアを得た．また，ROUGE-3, 4 においては，SENNA の分散表現を基に学習した手法 (SENNA-Opt) のスコアが最も高かった．

[3] https://code.google.com/p/word2vec/
[4] http://ml.nec-labs.com/senna/
[5] http://www.nltk.org/
[6] ROUGE の公式評価スクリプト (バージョン 1.5.5) においてオプション”-a -m -n 4 -2 4 -u -l 100 -c 95 -r 1000 -f A -p 0.5 -t 0 -x”で実行した際のスコアである．
[7] https://github.com/olofmogren/multsum

表 1: ROUGE による評価結果: ROUGE の各指標において最もスコアの高かったものを太字で示している．また，word2vec-Opt および SENNA-Opt においては分散表現の最適化を行なった結果，初期値として用いた word2vec および SENNA よりもどの程度 ROUGE スコアが向上しているかを数値と共に記載している．

	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-4	ROUGE-SU4
LexRank	53.52	17.62	5.92	2.04	24.71
TF-IDF	52.85	16.39	5.41	1.69	23.61
Embedding	53.72	17.55	6.10	2.01	24.52
Cosine (W2V)	53.12	17.27	5.35	1.45	24.72
Cosine(SENNA)	53.95	17.45	5.74	2.06	24.89
提案手法 Cosine(W2V-Opt)	**56.22**(+3.10)	**18.98**(+1.71)	6.59(+1.24)	2.27(+0.82)	**26.76**(+2.04)
提案手法 Cosine(SENNA-Opt)	55.15(+1.20)	18.53(+1.08)	**6.88**(+1.14)	**2.59**(+0.53)	26.04(+1.15)

結果として，単語分散表現の最適化を行なったものは，元の初期値である word2vec および SENNA を用いたものと比較して全ての ROUGE の評価尺度においてスコアを向上することができた．

6 おわりに

本論文では，分散表現を用いた文書要約における単語分散表現の学習手法の提案を行なった．レビュー要約のデータセットを用いて評価した結果，単語分散表現の学習を行なうことで，元の訓練済みの単語分散表現を用いるよりも高い ROUGE スコアを記録し，文書と参照要約の分散表現の距離を最小化させるというモデルが有効なことを確認した．

今後は，各文の ROUGE 値を予測し，最大になるものを逐次的に要約に加える教師あり手法との比較を行うほか，文書全体ではなく文単位での分散表現の学習を行なった場合どの程度影響があるのか調べたい．

謝辞 本研究は一部科研費基盤研究 (B)15H02749 の支援を受けて行った．

参考文献

[1] Z. Cao, F. Wei, S. Li, W. Li, M. Zhou, and H. Wang. Learning summary prior representation for extractive summarization. In *ACL*, pages 829–833, 2015.

[2] G. Erkan and D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479, 2004.

[3] K. Ganesan, C. Zhai, and J. Han. Opinosis: a graph-based approach to abstractive summarization of highly redundant opinions. In *COLING*, pages 340–348. Association for Computational Linguistics, 2010.

[4] M. Kågebäck, O. Mogren, N. Tahmasebi, and D. Dubhashi. Extractive summarization using continuous vector space models. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality*, pages 31–39. Citeseer, 2014.

[5] H. Kobayashi, M. Noguchi, and T. Yatsuka. Summarization based on embedding distributions. In *EMNLP*, pages 1984–1989, 2015.

[6] R. Lebret and R. Collobert. Word embeddings through hellinger PCA. In *EACL*, pages 482–490, 2014.

[7] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*, volume 8. Barcelona, Spain, 2004.

[8] H. Lin and J. Bilmes. A class of submodular functions for document summarization. In *ACL*, pages 510–520. Association for Computational Linguistics, 2011.

[9] P. Liu, S. R. Joty, and H. M. Meng. Fine-grained opinion mining with recurrent neural networks and word embeddings. In *EMNLP*, pages 1433–1443, 2015.

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

[11] O. Mogren, M. Kågebäck, and D. Dubhashi. Extractive summarization by aggregating multiple similarities. In *Proceedings of Recent Advances in Natural Language Processing*, pages 451–457, 2015.

[12] A. Nenkova and K. McKeown. Automatic summarization. *Foundations and Trends in Information Retrieval*, 5(2-3):103–233, 2011.

[13] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, Stanford University, Stanford, CA, USA, 1998.

[14] P. Ren, F. Wei, Z. CHEN, J. MA, and M. Zhou. A redundancy-aware sentence regression framework for extractive summarization. In *COLING*, pages 33–43, 2016.