言語概念ベクトルを用いた文書間類似度に基づく複数文書自動要約

住田 恭平† 二宮 崇††
†愛媛大学 工学部 情報工学科
††愛媛大学 大学院理工学研究科 電子情報工学専攻
{sumida@ai., ninomiya@}cs.ehime-u.ac.jp

1 はじめに

複数文書要約は同じ題目について記述されている複数の文書に対して単一の要約文書を生成するタスクである．近年，複数文書要約問題を整数線形計画問題 (ILP) に定式化し，その近似解を求めるか，ILP ソルバーを用いて厳密解を求めることにより要約の問題を解く手法が提案され，非常に高い要約精度を実現している．要約問題を ILP に定式化する手法としては，大別して，文の関連度や冗長度に基づく定式化 [1] と，被覆される単語の重要度に基づく定式化 [2, 3] に分けられる．本研究は，前者の手法のうち McDonald による手法に基づき，文書間の関連度を最大化する ILP を定式化し，言語概念ベクトル間の近さおよび要約中の文数の逆数を文書間の関連度とする要約手法を提案する．言語概念ベクトルとしては，word2vec と呼ばれる語彙に対する分散表現 [4] と paragraph vector と呼ばれる文書に対する分散表現 [5] を用いる．Mikolov らの手法によって比較的容易に高精度の言語概念ベクトルを取得することができるため，これらを用いることにより，文書間関連度の最大化に基づく ILP による要約手法が高精度化されることを実験により示す．

2 関連研究

Carbonell と Goldstein らは Maximum Marginal Relevance (MMR) と呼ばれる基準に従って要約文を抽出する手法を提案した [6, 7]．要約対象の文書 D が与えられたとき，文書中の各文 s に対する MMR は，D と s の関連性のスコアと，すでに選択された要約文集合と s との間の冗長性のスコアとの差により求められる．MMR に基づく要約アルゴリズムは MMR の高い順に要約文を選択する貪欲アルゴリズムとなっているが，McDonald [1] は MMR のアイデアを発展させ，制約付き最適化問題として次のように定式化し，要約文集合に対する最適な解を得る手法を提案した．

$$S = \arg\max_{S \subseteq D} \sum_{s \in S} Rel(s) - \sum_{s,t \in S} Red(s,t) \quad \text{s.t.} \sum_{s \in S} len(s) \leq K$$

ただし，D は要約対象の文書，S が得られる要約の文集合，$Rel(s)$ は文 s と D の関連性を示すスコア，$Red(s,t)$ は文 s と t の間の冗長性を示すスコア，$len(s)$ は文 s の長さ，K は要約の長さ制限にあたる要約最大長である．McDonald は，文書 D や各文 s を TFIDF による文書ベクトルで表現し，関連性のスコア $Rel(s)$ を，s の文書中での位置の逆数と文書 D と s とのコサイン類似度の和とし，$Red(s,t)$ を s と t のコサイン類似度として実験を行った．この問題は式を変形することにより整数線形計画問題 (ILP) とすることができ，ILP ソルバーを用いることで最適解を求めることができる．

McDonald はさらに各文と文書との関連性の合計ではなく，選択された要約文集合に対する最適化問題を次のように定式化し，要約文書と要約対象の文書間のコサイン類似度を最大化することにより要約文書を求める手法を提案している [1]．

$$S = \arg\max_{S \subseteq D} SIM(S,D) \quad \text{s.t.} \sum_{s \in S} len(s) \leq K$$

ただし $SIM(S,D)$ は S と D の類似度である．この問題は一般には ILP とはならないため，貪欲アルゴリ

[1] 正確にはクエリー付複数文書要約問題に対する定式化となっており，Q をクエリーとしたとき，$S = \arg\max_{S \subseteq D} SIM(S,Q) + SIM(S,D)$ である．

ズムや動的計画法を用いて近似解を求めている．本研究は McDonald のこの手法に基づいているが，本研究ではコサイン類似度の代わりに内積や要約に採用された文の数を用いて ILP としているところが異なる．

要約の問題を ILP とする手法は大別して，文の関連度や冗長度に基づく定式化 [1] と，被覆される単語の重要度に基づく定式化 [2, 3] がある．McDonald による手法は前者の定式化であり，本研究の提案手法も前者の定式化に基づく．前者の定式化では，文書のベクトルを用いて文書間の近さを表すため，単語埋め込みやパラグラフ埋め込みにより得られる言語概念ベクトルを用いる手法に拡張することが容易である．

言語概念ベクトルを獲得する手法として，本研究では，Mikolov らによる word2vec [4] と Le らによる paragraph vector [5] を用いた．word2vec は，文書中のある単語を隠した上でその単語を予測する擬似的なタスクを解くニューラルネットワークを大量の文書から学習し，各単語に対する重みベクトルを単語に対する概念ベクトルとする手法である．paragraph vector は word2vec を拡張し，パラグラフ ID に対する中間層を設けることによりパラグラフに対する概念ベクトルを獲得する手法である．

3 提案手法

本稿では言語概念ベクトルと整数線形計画法を用いた文選択による複数文書自動要約を扱う．要約対象の文書から文書の言語概念ベクトル (word2vec や paragraph vector) を求め，要約前の文書と要約後の文書の類似度を最大化する ILP を解くことで最適な要約文書を生成する．この際に文を言語単位とすることで生成される要約文書の文法性が保証される．本研究では文書間の類似度を内積と要約に採用された文の数を用いて定義する．$\vec{d}$ を要約前の文書のベクトルとし，$\vec{e}$ を要約後の文書のベクトルとする．文書は文の集合であると考え，文書全体の概念ベクトルは文の概念ベクトルの合計で表すこととした．つまり，x_i を i 番目の文が選択された場合に 1 となり、選択されない場合に 0 となる変数とし，$\vec{v_i}$ を要約前の文書の i 番目の文の概念ベクトルとすると，次のように表される．

$$\vec{d} = \sum_i \vec{v_i}$$

$$\vec{e} = \sum_i \vec{v_i} x_i$$

$\vec{d}$ は変数を含まないので定数ベクトルとして扱うことができる．

McDonald による手法では下記のようにコサイン類似度を最大化する問題となる．

$$max. \quad \frac{\vec{d}\cdot\vec{e}}{|\vec{d}||\vec{e}|} \tag{1}$$

$$s.t. \quad \sum_i l_i x_i \leq K, \tag{2}$$

$$x_i \in \{0,1\}; \forall i, \tag{3}$$

ここで l_i は i 番目の文の長さを表している．式 (1) において $\vec{d}$ と $\vec{e}$ のコサイン類似度を最大化することにより，2 つの文書の類似度を最大化する．式 (2) によって要約文の長さが K 以下であるという制約を与える．しかし，式 (1) において $|e|$ を ILP として正確に定式化することは困難である．

そのため本研究ではコサイン類似度を最大化するのではなく内積を最大化する問題を解く．また，このままでは文の長さが短いものばかりを採用すると考えられる．これを防ぐために，内積を要約に採用された文数で割る．以上の変更に加え，各ベクトルの表現を変更したものを以下に示す．

$$max. \quad \frac{\vec{d}\cdot\sum_i \vec{v_i} x_i}{\sum_i x_i} \tag{4}$$

$$s.t. \quad \sum_i l_i x_i \leq K, \tag{5}$$

$$x_i \in \{0,1\}; \forall i, \tag{6}$$

このままでは分母に変数が含まれるため，$t = 1/\sum_i x_i$，$y_i = t x_i$ と置き，以下のように式変形を行うことで整数線形計画問題として定式化する [8].

$$max. \quad \vec{d}\cdot\sum_i \vec{v_i} y_i \tag{7}$$

$$s.t. \quad \sum_i l_i y_i \leq Kt, \tag{8}$$

$$0 \leq y_i \leq 1 \quad ;\forall i, \tag{9}$$

$$0 \leq t \leq 1 \tag{10}$$

4 実験

4.1 実験設定

言語概念ベクトルを取得する方法として，1-of-k 表現，word2vec[4]，paragraph vector[2][5] を用いた．1-of-k 表現，word2vec においては単語単位でベクトルが取得されるため，一文中に出現する単語のベクトルの合計を単語の数で割ったものをその文のベクトルとした．ただし，文を単語の集合と考えた際に，ストップワードを用いることは有効ではないと考え，ストップワードリスト (ROUGE で用いられているもの) に記載されてい

[2] https://github.com/klb3713/paragraphvector(2016/01/10 アクセス)

表 1: 1-of-k ベクトルの ROUGE-1 値

1-of-k	
正規化	非正規化
0.2843	0.1644

表 2: word2vec の ROUGE-1 値

	ltw_eng+DUC'04		DUC'04	
次元数	正規化	非正規化	正規化	非正規化
50	0.3015	0.2980	0.2445	0.2475
100	0.3045	0.3027	0.2449	0.2478
200	0.3071	0.3039	0.2455	0.2448
500	0.3072	0.3012	0.2063	0.2460
800	0.3067	0.3017	0.1560	0.2536

る単語は除外した．word2vec，paragraph vector の学習用コーパスとしては DUC'04[9] の task2 のデータセットと Gigaword corpus の Los Angeles Times/Washington Post Newswire Service(ltw_eng) を使用した．なお，学習の際には，python の string モジュールの punctuation を利用して句読点を取り除いている．学習の次元数としては，50，100，200，500，800 次元を設定しそれぞれを比較する．また，各文のベクトルに対して正規化した場合と正規化しない場合を評価した．DUC'04 の task2 の設定に合わせ，要約のサイズは 665 バイト以下とした．評価には ROUGE version 1.5.5[3][10, 11] を用い，オプションは DUC'04 の task2 の設定から，-a -c 95 -b 665 -m -n 4 -w 1.2 とした．特に ROUGE-1 を用いた．また，ILP ソルバーとして ILOG CPLEX Ver. 12.5.1 (IBM 社) を用いた．

4.2 実験結果

表 1，表 2[4]，表 3 に実験結果を示す. また，提案手法の要約精度と文書要約の分野において高い精度を出していることで知られている高村ら [3] の手法および DUC'04 において最も高い ROUGE-1 値を記録した peer65 の手法の要約精度の比較について表 4 に示す. 表 1，2，3 において下線を引いてあるものはその手法において最も高い ROUGE-1 値を出したものである．

[3] http://www.berouge.com/Pages/default.aspx(2016/01/10 アクセス)

[4] 正規化した次元数 500，800 の word2vec と DUC'04 を用いた場合において精度が極端に低下しているが，それぞれの要約文書を確認したところ，幾つかのデータにおいて要約文書が生成されていないものがあった．これは文抽出プログラムの欠陥であると考えられる．

表 3: paragraph vector の ROUGE-1 値

	ltw_eng+DUC'04		DUC'04	
次元数	正規化	非正規化	正規化	非正規化
50	0.3153	0.3380	0.3047	0.3316
100	0.3162	0.3454	0.3091	0.3330
200	0.3213	0.3423	0.3075	0.3362
500	0.3259	0.3559	0.3052	0.3278
800	0.3268	0.3495	0.3028	0.3346

表 4: 提案手法 (正規化していない次元数 500 の paragraph vector と ltw_eng+DUC'04 を用いた場合) と高村らの手法と peer65 の手法との比較

手法	ROUGE-1 値
提案手法	0.3559
高村らの手法	0.390
peer65	0.382

表 1，2，3 を見ると，必ずしも次元数を増加させれば精度が向上するとは限らないことがわかる．しかし，すべての条件下において学習用コーパスの量を増加させると要約の精度が向上している．加えて，すべての場合において paragraph vector の結果が上回っていることがわかる．また，表 2，3 を見ると正規化をすることで要約の精度が低下し，paragraph vector の結果においては精度が 2%以上低下している．最も高い ROUGE-1 値は paragraph vector を用い，次元数を 500 次元に設定し取得されたベクトルを正規化しない場合で 0.3559 であった．

表 1，2，3 から，概念ベクトルを用いた要約文書生成においては paragraph vector を用いて概念ベクトルを取得する手法が 1-of-k ベクトルと word2vec を用いる手法に比べて優れていると考えられる．これは文ベクトルを単語ベクトルの加算で表現する手法では，文のコンポジショナリティを十分に再現できていないからではないかと考える．

また，word2vec と paragraph vector を用いた結果では正規化をすることで精度が全体的に低下している。これは word2vec と paragraph vector によって生成されるベクトルの大きさには要約において重要な情報が含まれているためと考えられる．

5 まとめ

本研究は，McDonald の手法に基づき，文書間の関連度を最大化する ILP を定式化し，言語概念ベクトル間の近さおよび要約中の文数の逆数を文書間の関連度とする要約手法を提案した．言語概念ベクトルとしては，word2vec と呼ばれる語彙に対する分散表現と paragraph vector と呼ばれる文書に対する分散表現を用い，paragraph vector を用いる手法が最も高い精度を実現した．

本研究では内積の最大化によって要約文書を生成したが，コサイン類似度を求めることができればより精度が向上すると考えられる．または Jaccard 係数や Dice 係数を使った類似度計算を ILP として定式化することができればより良い精度を求めることができるであろう．

謝辞

本研究は JSPS 科研費 25280084 の助成を受けたものである。ここに謝意を表する。

参考文献

[1] Ryan McDonald. A study of global inference algorithms in multi-document summarization. In *Proceedings of the 29th European Conference on IR Research*, ECIR'07, pp. 557–564, Berlin, Heidelberg, 2007. Springer-Verlag.

[2] Dan Gillick and Benoit Favre. A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing*, ILP '09, pp. 10–18, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[3] 高村大也, 奥村学. 最大被覆問題とその変種による文書要約モデル. 人工知能学会論文誌, Vol. 23, No. 6, pp. 505–513, 2008.

[4] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In C.J.C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pp. 3111–3119. Curran Associates, Inc., 2013.

[5] Quoc V. Le and Tomas Mikolov. Distributed representations of sentences and documents. *CoRR*, Vol. abs/1405.4053, , 2014.

[6] Jaime Carbonell and Jade Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pp. 335–336, New York, NY, USA, 1998. ACM.

[7] Jade Goldstein, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. Multi-document summarization by sentence extraction. In *Proceedings of the 2000 NAACL-ANLPWorkshop on Automatic Summarization - Volume 4*, NAACL-ANLP-AutoSum '00, pp. 40–48, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[8] 梅谷俊治. 組合せ最適化入門：線形計画から整数計画まで. 自然言語処理, Vol. 21, No. 5, pp. 1059–1090, 2014.

[9] DUC. Document understanding conference. In *HLT/NAACL Workshop on Text Summarization*, 2004.

[10] Chin-Yew Lin and Eduard Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pp. 71–78. Association for Computational Linguistics, 2003.

[11] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In Stan Szpakowicz Marie-Francine Moens, editor, *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pp. 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.