引用済み論文の情報を用いた引用論文推薦

田中陸斗 [1] 杉山弘晃 [2] 平博順 [3] 有田朗人 [3] 堂坂浩二 [1]
[1] 秋田県立大学　[2]NTT コミュニケーション科学基礎研究所 [3] 大阪工業大学
{b20p025, dohsaka}@akita-pu.ac.jp, h.sugi@ieee.org
{hirotoshi.taira, m1m21a02}@oit.ac.jp

概要

近年の爆発的な学術論文数の増加により，引用論文推薦のニーズが高まっている．本稿では，関連研究セクションに着目し，セクション内に既にいくつかの引用が付与されている状況を想定した引用論文推薦手法を提案する．本手法は，引用を付与すべき箇所 (引用マーカー) が与えられたときに，引用マーカー前後の単語 (引用コンテキスト) と引用済み論文の情報を使って引用すべき論文を推薦する．手法は 2 つのフェーズから成る．フェーズ 1 では，事前に，引用コンテキストと対応する引用論文の距離が近くなるように SentenceBERT のモデルを学習する．このモデルを使って，与えられた引用コンテキストと距離の近い論文をいくつか候補とする．フェーズ 2 では，事前に，同じ論文内で引用されている論文同士の距離が近くなるように SentenceBERT のモデルを学習しておき，このモデルと引用済み論文の情報を使って，フェーズ 1 の候補論文をさらに適した順位になるようにリランキングする．フェーズ 2 によって，フェーズ 1 に比べて論文推薦の Recall, MRR の値がともに上昇することが示された．

1　はじめに

学術論文を執筆する際，論文中の主張を裏付けるために適切な引用を行うことは重要である．しかし，近年論文数が爆発的に増加し，引用すべき適切な論文を見つけることは非常に労力のかかる作業となっている．こうした中，研究者の論文執筆支援を目的とした研究が進められてきた．成松ら [1, 2] は，研究者の論文執筆における関連研究の引用および生成に関わる統合的な執筆支援を目的として, 関連研究に関わる様々な既存のタスクを統合した新たなデータセット構築方法および 5 つのタスクを定義した．本稿では，その中の引用論文推薦タスクに着目する．これは与えられたテキストに対して適切な引用論文を推薦するタスクであり，大域的引用論文推薦と局所的引用論文推薦に分類される [3]．大域的引用論文推薦では，論文本文全体または要旨を入力するのに対し，局所的引用論文推薦では，引用コンテキストと呼ばれる一文，あるいは単語列を入力する．本稿では，局所的引用論文推薦のタスクを扱う．表 1 に例を示す．入力は [4] より引用した．[X] は引用を付与すべき場所であり，これを引用マーカーと呼ぶ．この例では，引用コンテキストを [X] を含む一文と定義し，[X] に入る引用論文を推薦している．

表 1　局所的引用論文推薦の入力と出力の例

入力	This is a good speedup trick because common words are accessed quickly. This use of binary Huffman code for the hierarchy is the same with [X].
出力	1. Distributed representations of phrases and their compositionality(2013) 2. Efficient estimation of word representations in vector space(2013)

本稿では，関連研究セクションに着目し，セクション内で既にいくつかの引用が付与されている状況を想定する．セクション内で 1 つの引用マーカーが与えられるとき，その引用マーカーに入るべき引用論文を，引用コンテキストと既に引用されている論文の情報を用いて推薦する手法を提案する．本手法では，引用コンテキストや論文の埋め込み表現を求める際，文の埋め込み表現の構築に特化した BERT である SentenceBERT（以下 SBERT）[5] を用いる．

局所的引用論文推薦の従来研究として，杉本ら [6] は，引用コンテキストと推薦する候補の論文の文章の双方を独立に BERT[7] で埋め込み，候補の論文をコサイン類似度でランク付けするモデルを提案しているが，この研究では主に引用コンテキスト

に焦点を当てており，いくつか引用が付与されている状況は想定していない．また，候補の論文はBM25[8] を用いて上位 2,048 本の論文としているが，本手法では BM25 を用いていないため，性能の比較は行えない．Zhang ら [9] は，原稿の一部に適切な引用が付与されている状況を想定した引用論文推薦を行っている．文書の特徴ベクトルを表現できるように Word2Vec[10] を拡張したアルゴリズムである Paragraph Vector[4] を使用して文書の埋め込み表現や，引用済みの論文の埋め込み表現を計算して引用論文推薦に取り入れている．これに対し，提案手法では，引用コンテキストや論文の埋め込み表現を求める際，従来研究で用いられた BERT や Paragraph Vector に代わり，SBERT を用い評価を行った．

以下において，2 節で提案手法を示す．3 節で，データセット並びに評価方法を説明し，評価結果について考察する．最後に，4 節で結果についてまとめる．

2 提案手法

本手法は，与えられた引用コンテキストに対して，引用すべき候補論文をいくつか出力するフェーズ 1 と，既に引用されている論文の情報を使って，候補論文をさらに適した順位に並び替えるフェーズ 2 から構成される．

2.1 フェーズ 1: 候補選択

このフェーズでは，SBERT を使用し，引用コンテキストと対応する引用論文（以下，正例）の距離が，それ以外の論文（以下，負例）の距離よりも近くなるように，引用テキストと論文の埋め込み表現を学習する．論文の入力は，タイトルとアブストラクトを結合させたものとした．損失関数には，以下の式で表されるトリプレット損失関数を使用した [11]．

$$Loss = max\{(||C - P^{+}|| - ||C - p^{-}|| + \epsilon), 0\} \quad (1)$$

ここで，C は引用コンテキストの埋め込み，p^{+} は正例の埋め込み，p^{-} は，負例の埋め込みを示す．負例はランダムに選択した．また，元論文 [5] に従い，$\|\cdot\|$はユークリッド距離を使用し，マージン ϵ は 1 とした．

フェーズ 1 では，このモデルを用いて，引用コンテキストに対応する候補論文を k 件取得する．以下の手順にしたがって推論を行う．

1. 引用コンテキストをモデルに入力し，埋め込みを得る．
2. 論文のタイトルとアブストラクトを結合したものをモデルに入力し，埋め込みを得る．
3. 1 の埋め込みと，2 の埋め込みのコサイン類似度を計算する．

引用コンテキストと論文プール内の各論文とのコサイン類似度を計算し，高い順に k 件の論文を取得し，その引用コンテキストの候補とする．

2.2 フェーズ 2: リランキング

このフェーズでは，SBERT を用いて，同じ関連研究セクション内で引用されている論文同士の距離が，それ以外の論文との距離よりも近くなるような埋め込み表現を学習する．フェーズ 1 では引用コンテキストと論文の距離を学習していたが，このフェーズでは論文同士の距離を学習する．論文の入力には，タイトルと要旨を結合させたものを使用する．

学習時の一例として，ある関連研究セクション内に論文 a, b, c が引用されている場合を考える．a に着目すると，a と b および a と c の距離がそれぞれランダムに選んだ論文の距離よりも近くなるように学習を行う．今回の例では，得られる組み合わせは (a,b)，(a,c)，(b,c) の三組である．

損失関数にはトリプレット損失関数を使用した．フェーズ 1 と同様に，距離はユークリッド距離を使用し，ϵ の値は 1 とした．このモデルを用いて，フェーズ 1 で得られた k 件の候補論文をリランキングする．

推論の概要を図 1 に示す．この図は，同じ関連研究セクション内で引用されている論文を A，B，C とし，マスクした箇所に引用すべき候補論文を，フェーズ 1 で d_1，d_2，d_3，d_4 の 4 件取得した例である．以下の手順にしたがって推論を行う．

1. 関連研究セクション内でマスクされてない被引用論文（図では A,B,C）のタイトルと要旨を結合させたものをそれぞれモデルに入力し，埋め込みを得る．
2. フェーズ 1 で取得した各候補論文（図では d_1~d_4）のタイトルと要旨を結合させたものをそれぞれモデルに入力し，埋め込みを得る．
3. 1 の各埋め込みと，2 の各埋め込みのコサイン類似度を計算し，一番高いスコアをその候補論文のスコアとする．

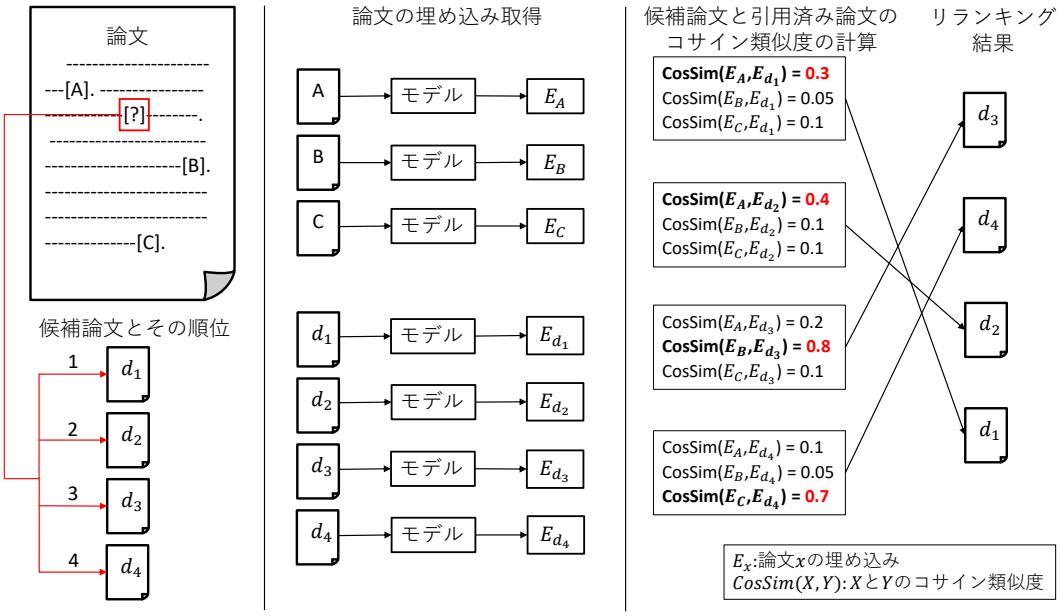

図 1　フェーズ 2 の概要

最後に，各候補論文を 3 のスコアが高い順に並び替えて，上位 k 件を推薦する．図では，d_1，d_2，d_3，d_4 の順位だったが，リランキングを行うことで d_3，d_4，d_2，d_1 の順位に変更されている．

3　実験

3.1　データセット

研究者の学術論文の執筆支援を目的として，小山ら [12] によって作成されたデータセットを使用する．このデータセットには，ArXiv から取得した論文の関連研究セクションが約 3 万件と，セクション内の被引用論文のタイトル，要旨が含まれている．被引用論文数の平均は 2.6 件で，最大値は 43 件である．また，引用コンテキストは引用マーカーから前後 50 単語を結合させたものと定義する．

全関連研究セクションのうち，引用論文数が 2 件未満のものを削除した約 13000 件を使用する．これを訓練データ，検証データ，テストデータにそれぞれ約 10400，1300，1300 ずつに分割する．また，引用コンテキストの数は訓練データ，検証データ，テストデータそれぞれで約 77000，9800，9500 である．このうち，フェーズ 1 で使用するテストデータの数は，関連研究セクション 1 つにつき 1 件の被引用論文をマスクするため，関連研究セクション数と同じ約 1300 件となる．論文プールの総数は約 38000 件である．

3.2　評価手法

提案した論文推薦システムが既存の論文の引用をどの程度予測できるかを測定する．本稿では，推薦された候補の上位 5 件と上位 10 件に対して Recall と MRR を計算して評価する．Recall の計算方法を以下に示す．

$$Recall@k = \frac{|\alpha \cap p_k|}{|\alpha|} \tag{2}$$

ここで，k は考慮する上位ランキングの数，α は正解の被引用論文集合，p_k は上位 k 件の推薦リストである．

また，MRR の計算方法を以下に示す．

$$MRR = \frac{1}{|U|}\sum_{u \in U}\frac{1}{k_u} \tag{3}$$

ここで，u はあるテストデータ，U はテストデータ全体，k_u はテストデータ u の推薦リストのうち，最初に u の正解の論文が現れた順位である．

3.3　結果と考察

フェーズ 1 を使用した場合の結果を表 2 に示す．この結果は，フェーズ 1 において，引用コンテキスト 1 件に対して負例の数を変化させた場合の性能の比較を行い，Recall@10 で最も性能がよかったものを採用した．負例の変化による性能を比較したものを結果を表 3 に示す．負例を増やすことで性能が上昇していることが確認できるが，負例の数が 7 件を

表 2　フェーズ 1 単体での結果

	@5		@10		@30		@50		@80	
	MRR	Recall	MRR	Recall	MRR	Recall	MRR	Recall	MRR	Recall
フェーズ 1	0.258	0.350	0.274	0.464	0.284	0.628	0.286	0.709	0.287	0.768

表 3　フェーズ 1: 負例の数による性能の比較

負例の数	MRR	Recall@5	MRR	Recall@10
1	0.193	0.274	0.205	0.356
3	0.218	0.301	0.232	0.400
5	0.257	**0.351**	0.272	0.455
7	**0.258**	0.350	**0.274**	**0.464**
10	0.257	0.348	0.271	0.463

超えるとほぼ変化が見られない．また，負例の数が 10 件になるとスコアが低下していることから，負例の数は 7 件で学習が十分であり，これ以上の負例の増加は過学習を起こす可能性があると考えられる．

フェーズ 1 の上位 5 件と上位 10 件までの性能が，フェーズ 2 で向上するか否かで，提案手法の有効性を示す．フェーズ 2 を用いて，フェーズ 1 で取得した候補論文をリランキングした結果を表 4 に示す．取得した候補論文数により，性能が変化することが確認できる．最も性能が良かったのは候補論文数が 80 件のときだが，それ以外でもフェーズ 1 の結果を上回っている．このことから，既に引用されている論文の情報を使用することが有効に働くことが分かった．

4　おわりに

本稿では，関連研究セクション内にいくつか引用が付与されている状況を想定した引用論文推薦手法を提案した．この手法は，関連研究セクション内で，引用を付与すべき 1 つの引用コンテキストが与えられるとき，引用すべき論文を，2 つのフェーズに分けて推薦する．フェーズ 1 では，SBERT を用いて，引用コンテキストに適切な引用すべき候補論文をいくつか出力した．フェーズ 2 では，候補論文を既に引用されている論文の情報を使って，さらに適した順位に並び替えた．フェーズ 2 のリランキングにより，フェーズ 1 単独の場合と比べて性能が良くなったことから，引用済み論文の情報を用いることにより引用論文推薦の性能が向上することが示された．

今後の課題として，提案手法と同様に，引用済み論文の情報を使用した論文推薦の従来手法 [9] と比較し，従来手法で使った Paragraph Vector と本手法で使った SBERT の効果の差異を考察することがある．その際，従来手法は本稿で使用したものと異なるデータセットを使っているため，従来研究のデータセットに揃えて提案手法と比較したい．また，本手法では，論文の情報としてタイトルと要旨のみを使用しているが，著者などの他の情報を取り入れる手法の検討も考えられる．

表 4　フェーズ 2: フェーズ 1 で取得した候補論文数による性能の比較

候補論文数	MRR	Recall@5	MRR	Recall@10
30	0.311	0.388	0.322	0.477
50	0.316	0.393	0.328	0.490
80	**0.317**	**0.394**	**0.330**	**0.496**

謝辞

本研究の遂行にあたり，ご助言・ご協力をいただきました，電気通信大学小山康平氏，NTT コミュニケーション科学基礎研究所成松宏美氏，電気通信大学南泰浩教授，工学院大学大和淳司教授，名古屋大学東中竜一郎教授，農研機構菊井 玄一郎チーム長に感謝いたします．また，日頃より丁寧にご指導してくださった秋田県立大学石井雅樹准教授，伊東嗣功助教に感謝いたします．

参考文献

[1] 成松宏美, 小山康平, 堂坂浩二, 田盛大悟, 東中竜一郎, 南泰浩, 平博順. 学術論文における関連研究の執筆支援のためのタスク設計およびデータ構築. 言語処理学会第 26 回年次大会, 2021.

[2] Hiromi Narimatsu, Kohei Koyama, Kohji Dohsaka, Ryuichiro Higashinaka, Yasuhiro Minami, and Hirotoshi Taira. Task Definition and Integration for Scientific-Document Writing Support. In Proceedings of the 1st Workshop on Scholarly Document Processing, 2021.

[3] Michael Färber and Adam Jatowt. Citation recommendation: approaches and datasets. **International Journal on Digital Libraries**, Vol. 21, pp. 375–405, 2020.

[4] Q. V. Le and T. Mikolov. Distributed representations of sentences and documents. ICML, 2014.

[5] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In Proceedings of Empirical Methods in Natural Language Processing (EMNLP), 2019.

[6] 杉本海人, 相澤彰子. BERT-based Bi-Ranker による文

脈を考慮した引用論文推薦. 言語処理学会第 26 回年次大会, 2021.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding,. In Proceedings of NAACL-HLT, 2019.

[8] Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond. **Foundations and Trends in Information Retrieval**, Vol. 3, No. 4, pp. 333–389, 2009.

[9] Y. Zhang and Q. Ma. Doccit2vec: Citation recommendation via embedding of content and structural contexts. IEEE, 2020.

[10] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781, 2013.

[11] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning Fine-grained Image Similarity with Deep Ranking. In Proceedings of CVPR, 2014.

[12] 小山康平, 南泰治, 成松宏美, 堂坂浩二, 東中竜一郎, 田盛大悟, 平博順. 学術論文における関連研究の執筆支援のための被引用論文の推定. 言語処理学会第 26 回年次大会, 2021.