中間タスクの挿入による学術論文における URL 引用の分類

和田 和浩 [1]　松原 茂樹 [1,2]
[1] 名古屋大学情報学部　[2] 名古屋大学情報連携推進本部
{wada.kazuhiro.s8@s,matsubara.shigeki.z8@f}.mail.nagoya-u.ac.jp

概要

学術活動において，データやプログラムなどの学術資源にアクセスできることは重要である．特に URL による引用（以下，URL 引用）はその論文で使用された学術資源を参照していることが多く，これらを分類し利用することで学術資源へのアクセス性を高めることができる．そこで本論文では，学術論文における URL 引用の種類と目的でクラス分類する手法について述べる．本手法では，メインタスクで使用するデータを利用し，その入力の対を用いた中間タスクを設定し導入した．分類実験の結果，先行研究に対する本手法の優位性を確認した．

1　はじめに

学術活動において，学術論文，及び，そこで利用されたデータやプログラムなどの学術資源にアクセスできることは重要である．これらのアクセス性を向上させるための様々なデータベースやサービスが存在する．例えば，Google Scholar[1)]，Semantic Scholar[2)]では単なる論文の提示だけでなく，論文中の引用関係に基づき，関連する論文に辿ることができる．このように，引用関係を用いることは学術資源へのアクセス性を高める上で有用である．

これらのサービスが対象とする引用は，主に参考文献に記された書誌情報である．しかし，学術論文における引用対象は，参考文献リストの書誌情報以外に，URL による引用（以下，URL 引用）もあり，これらは参考文献リストに加えて脚注や本文中にも出現する．こうした URL 引用はその論文で使用されたデータやプログラムなどの学術資源を含むことが多い．URL 引用を利用することで論文中のデータやプログラムなどのアクセス性が高まる．また，URL による引用は種類や目的が多様であるため，それらを含めて提示することにより，サービスの利便性が向上することが期待できる．

これを実現するために，論文における URL 引用を種類と目的で分類する必要がある．引用の種類や目的の分類体系については既に様々な提案がある [1, 2, 3]．なかでも角掛らは，URL による引用についてその種類と目的のクラスを整理しており，SciBERT[4] を用いた分類手法を提案している [1]．しかし，角掛らの分類手法には，少数クラスに対する分類性能が低く，一部のクラスが予測結果として全く出力されない場合があった．

そこで本論文では，中間タスクを用いた論文における URL 引用の分類手法を提案する．中間タスクを用いた手法は既に多くの提案があり，その有効性が示されている [5, 6]．しかし，中間タスクの挿入には

- 中間タスクに必要なデータを新たに用意する必要がある
- 適切な中間タスクを選択する必要がある

という課題を解決する必要がある [7, 8]．

メインタスクに関連したタスクを選ぶことで適切な中間タスクの選択をする手法 [7] が提案されており，メインタスクと中間タスクとの関連性の高さは適切な中間タスクの選択に重要である．本手法ではメインタスクで使用するデータを再利用して，メインタスクにおける入力の対を用いた中間タスクを作成した．これにより，中間タスク用の新たなデータを必要とせずにメインタスクに関連した中間タスクの使用が可能となる．

本手法を評価するために実験を行い，先行研究の手法との分類性能を比較した．実験の結果，本手法の有効性を確認した．

1）Google Scholar　https://scholar.google.co.jp/
2）Semantic Scholar　https://www.semanticscholar.org/

2 関連研究

2.1 引用の分類

論文中の引用を分類する取り組みは既に存在しており，角掛らは学術論文中の URL が指すものについて，ツールとデータを区別する分類タスクに取り組んだ [9]．また，引用の目的を 6 種類のクラスに分類する Shared Task が提案されている [2]．このタスクに対する取り組みとして，Baig らは TF-IDF とデータを分析して作成した特徴量を用いた手法 [10] を，Maheshwari らは大規模言語モデルである SciBERT[11] を利用した手法 [4] をそれぞれ提案している．そして，角掛らは URL 引用に注目し，その URL が指すものが果たす役割と種類，引用の目的の分類体系を作成し，その分類手法を提案している [1]．

2.2 中間タスクの挿入

大規模言語モデルを用いた学習では通常，大規模なコーパスを用いて事前学習を行ったのち，メインタスクのファインチューニングを行う．これに対して事前学習とメインタスクのファインチューニングとの間に別のタスク（以下，中間タスク）を設け，そのファインチューニングを行うことで，メインタスクに有用な特徴を効果的に学習する方式が提案されている．Phang らは中間タスクの挿入により GLUE ベンチマークの性能が改善することを示している [5]．

その一方で，中間タスクを挿入すれば必ず効果をもたらすとは限らないという問題点も指摘されている [7, 8]．Poth らは中間タスク用のデータや学習済みモデルの利用可能性に応じて中間タスクを適切に選択する手法を提案している [7]．また Pruksachatkun らは特定の能力を問う probing task に対する性能を分析することで，中間タスクとして効果的なタスクの特徴について分析している [8]．

3 提案手法

3.1 問題設定

本研究では，角掛らが提案した URL 引用の種類と目的の分類 [1] に基づきアノテーションされたデータを使用する．1 つの URL 引用に対して，Role, Type, Function の 3 種類のラベルが存在する．Role, Type は URL が参照する学術資源の種類を表すラベルであり Role が大分類，Type が小分類に相当し，相互に対応関係がある．Function は URL 引用の目的を表すラベルである．本研究ではこれらの 3 つのラベルそれぞれの分類タスクに取り組む．

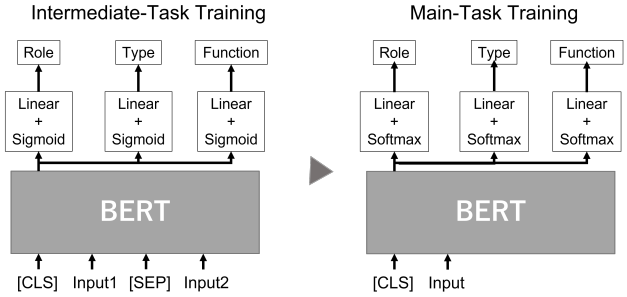

図 1 モデルの概略

3.2 手法の概要

提案手法の概略を図 1 に示す．提案手法では中間タスクとメインタスクに分けてモデルの学習を行う．いずれも BERT[12] を Encoder として利用して共有する．出力は中間タスク，メインタスクに共通して BERT の [CLS] トークンに対応する出力に対してラベルごとに全結合層を持つ．活性化関数として中間タスクではシグモイド関数を，メインタスクではソフトマックス関数を利用する．また，Role, Type, Function の各ラベル間の関連性を考慮して分類するため，マルチタスク学習を採用する [1, 13]．

3.3 中間タスクの学習

中間タスクの入力ではメインタスクにおけるモデルへの入力の対を使用し，出力では入力の対がメインタスクで同じクラスに属しているか否かの 2 値の分類を行う．このとき，Role, Type, Function ごとにクラスが割り当てられているため，3 種類のそれぞれでクラスが一致しているかどうかを判定する．損失関数として 3 つのラベルそれぞれの BCELoss の和を使用した．つまり，データセット中の i 番目のデータの入力を I_i，Role, Type, Function のラベルをそれぞれ R_i, T_i, F_i と記すとき，モデルの入力，出力及び損失関数は以下のようになる．

$$\text{入力：} I_{int} = [CLS] + I_i + [SEP] + I_j + [SEP] \quad (1)$$

$$\text{出力：} O_{role} = \sigma(W_{role} BERT(I_{int})) \quad (2)$$

$$O_{type} = \sigma(W_{type} BERT(I_{int})) \quad (3)$$

$$O_{function} = \sigma(W_{function} BERT(I_{int})) \quad (4)$$

表 1 Role のクラス分布

	標本数	割合
Method	1102	0.369
Material	870	0.291
補足資料	835	0.279
Mixed	182	0.061

表 2 Type のクラス分布

	標本数	割合		標本数	割合
Tool	629	0.210	Paper	279	0.093
Code	473	0.158	DataSource	235	0.079
Dataset	353	0.118	Document	217	0.073
Website	305	0.102	Mixed	182	0.061
Knowledge	282	0.094	Media	34	0.011

表 3 Function のクラス分布

	標本数	割合
Use	1231	0.412
Introduce	886	0.296
Produce	653	0.218
Compare	111	0.037
Extend	100	0.033
Other	8	0.003

損失関数：

$$\begin{aligned} & BCELoss(\delta_{R_i R_j}, O_{role}) \\ + \; & BCELoss(\delta_{T_i,T_j}, O_{type}) \\ + \; & BCELoss(\delta_{F_i,F_j}, O_{function}) \end{aligned} \quad (5)$$

ただし，σ はシグモイド関数を，δ はクロネッカーのデルタをそれぞれ表す．

3.4 メインタスクの学習

メインタスクの学習では Role, Type, Function のクラス分類を行う．基本的なモデル構造は中間タスクの学習のものと同様であり，変更点は最後の各ラベルに対応する全結合層がそれぞれのクラス数に対応した出力になる点，及び，多クラス分類であるため活性化関数にソフトマックス関数を使用している点である．また，BERT 部分のパラメータについては中間タスクの学習後のものを初期値として使用している．損失関数は各ラベルの CrossEntropyLoss の和とした．

4 実験

提案手法を評価するため，URL 引用の分類実験を行った．

4.1 実験の概要

使用データ URL 引用に Role, Type, Function の 3 種類のラベルが付与されたデータ [1] を使用した (3.1 節参照)．各ラベルのクラス分布を表 1,2,3 に示す．Role は各クラスがほぼ均等に分布しているものの，Type，Function は，クラス間の不均衡が大きく，分類が困難であることが予想される．

実験設定 提案手法を PyTorch[3] と Hugging Face[4] を利用して実装した．中間タスク，メインタスクで共有する BERT には事前学習済みの公開モデル[5] を使用した．

モデルへの入力は角掛らの手法 [1] を参考に，節タイトル，引用文とその前後 1 文（計 3 文），脚注文または参考文献を [SEP] トークンで区切ったものを使用した．

学習時，中間タスクでは patience を 2 に，メインタスクでは 5 に設定して Early Stopping を行った．以下に学習時に設定したパラメータを示す．

- 学習率（共通）：1e-5
- 最適化手法（共通）：Adam[14]
- バッチサイズ：16（中間タスク），32（メインタスク）
- 中間タスク用に作成するデータの数：300,000

比較手法 以下の 2 種類の手法を用いる．

- Zhao らが提案した SciResCLS[15]．引用文脈を使用してマルチタスク学習を行う
- 角掛らによる手法 [1]．入力の素性として節タイトル，脚注文または参考文献を追加してマルチタスク学習を行う

評価指標 使用するデータは分布の偏りが大きいため，少数クラスに対する分類性能を評価するためにマクロ平均の F1 値を採用する．

表 4 実験結果

	Role	Type	Function
Zhao ら [15]	0.683	0.485	0.409
角掛ら [1]	**0.758**	0.551	0.447
提案手法	0.750	**0.583**	**0.504**

4.2 実験結果

学習，検証データの分割時のシード値を変更して複数回実験を行い，それぞれのマクロ平均の F1 値を平均した結果を表 4 に示す．Role は角掛らの手法に若干劣る結果となったが，Type では 2~3%，Function では 5~7%の性能の向上が確認できた．

表 5 に提案手法と角掛らの手法 [1] のあるシード値で学習したときのクラスごとの F1 値を示す．標本数の少ない Media, Compare, Extend に対しても，提案手法ではいくつか予測されるようになった．ク

3） PyTorch.org https://pytorch.org/
4） Hugging Face https://huggingface.co/
5） google/bert_uncased_L-12_H-768_A-12 https://huggingface.co/google/bert_uncased_L-12_H-768_A-12

表 5　クラスごとの F1

Role	F1-score		Type	F1-score		Function	F1-score	
	角掛ら [1]	提案手法		角掛ら [1]	提案手法		角掛ら [1]	提案手法
Method	0.776	0.760	Tool	0.661	0.657	Use	0.780	0.803
Material	0.682	0.659	Code	0.609	0.655	Introduce	0.787	0.800
補足資料	0.764	0.689	Dataset	0.585	0.622	Produce	0.806	0.790
Mixed	0.837	0.864	Website	0.542	0.625	Compare	0.000	0.588
			Knowledge	0.297	0.275	Extend	0.000	0.167
			Paper	0.857	0.735	Other	0.000	0.000
			DataSource	0.324	0.438			
			Document	0.500	0.533			
			Mixed	0.800	0.844			
			Media	0.000	0.333			

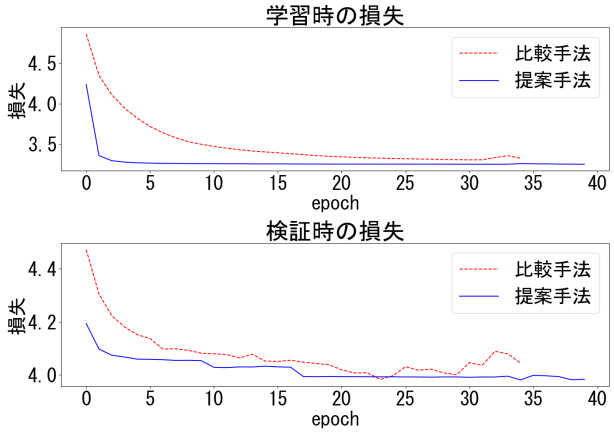

図 2　学習曲線

ラス分布の偏りが大きくなる Role，Type，Function の順に比較手法に対する性能の改善幅が大きくなっている．

4.3　考察

4.3.1　メインタスクの学習過程の比較

深層学習モデルの学習では，その性能だけでなく学習の安定性も重要である．そこで，学習及び検証時の損失関数の値の推移を確認した．図 2 に提案手法と角掛らの手法 [1] のメインタスクの学習時の損失の推移を示す．学習時，検証時ともに提案手法が角掛らの手法 [1] に比べて小さな損失となっている．また，角掛らの手法 [1] は検証時の損失に上下があるのに対して提案手法はエポック毎に安定して損失が減少しており，より安定した学習ができていることがわかる．

4.3.2　中間タスクの有効性

中間タスクで学習された特徴がメインタスクで有効に機能することを確認するために，中間タスクの学習後，BERT 部分のパラメータをフリーズさせてメインタスクの学習を行った．これにより，学習可能なパラメータは各ラベルの確率を出力するための全結合層 1 層となるため，この実験設定でもモデルの分類性能が維持されるならば，それは，中間タスクのみでファインチューニングされた BERT により，メインタスクを学習するために有効な特徴を表現できていることを示している．

表 6　BERT をフリーズさせたときの実験結果

	Role	Type	Function
Zhao ら [15]	0.683	0.485	0.409
提案手法 (フリーズなし)	0.702	**0.514**	0.472
提案手法 (フリーズあり)	**0.703**	0.490	**0.486**

実験結果を表 6 に示す．ただし，提案手法の入力は引用文のみとし，節タイトル，脚注文は使用していない．これは Zhao らの手法 [15] の入力と同様である．BERT 部分のパラメータをフリーズさせても性能は維持されており，中間タスクにより学習された特徴がメインタスクにおいて有効に働いている．

5　まとめ

本論文では，学術論文における URL 引用を種類と目的でクラス分類する手法を提案した．本手法では，メインタスクで使用するデータを利用し，その入力の対を用いた中間タスクを設定し導入した．そして，分類実験の結果，先行研究では性能が十分でなかった Type，Function において性能が向上しており，本手法の有効性を確認した．

今後の課題として，提案手法で性能の向上が確認できなかった Role に対する分類性能の改善に向け，複数の中間タスクを組み合わせて学習する手法を検討することがあげられる．

謝辞

本研究は，一部，科学研究費補助金（基盤研究（B））（No. 21H03773）により実施したものである．

参考文献

[1] Masaya Tsunokake and Shigeki Matsubara. Classification of url citations in scholarly papers for promoting utilization of research artifacts. In **Proceedings of the 1st Workshop on Information Extraction from Scientific Publications (WIESP) at AACL-IJCNLP**, November 2022.

[2] Suchetha N. Kunnath, David Pride, Drahomira Herrmannova, and Petr Knoth. Overview of the 2021 SDP 3C citation context classification shared task. In **Proceedings of the 2nd Workshop on Scholarly Document Processing**, pp. 150–158, Online, June 2021. Association for Computational Linguistics.

[3] David Jurgens, Srijan Kumar, Raine Hoover, Dan McFarland, and Dan Jurafsky. Citation classification for behavioral analysis of a scientific field, 2016.

[4] Himanshu Maheshwari, Bhavyajeet Singh, and Vasudeva Varma. SciBERT sentence representation for citation context classification. In **Proceedings of the 2nd Workshop on Scholarly Document Processing**, pp. 130–133, Online, June 2021. Association for Computational Linguistics.

[5] Jason Phang, Thibault Févry, and Samuel R Bowman. Sentence encoders on STILTs: Supplementary training on intermediate labeled-data tasks. November 2018.

[6] Ting-Yun Chang and Chi-Jen Lu. Rethinking why intermediate-task fine-tuning works. In **Findings of the Association for Computational Linguistics: EMNLP 2021**, pp. 706–713, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[7] Clifton Poth, Jonas Pfeiffer, Andreas Rücklé, and Iryna Gurevych. What to pre-train on? Efficient intermediate task selection. In **Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing**, pp. 10585–10605, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[8] Yada Pruksachatkun, Jason Phang, Haokun Liu, Phu Mon Htut, Xiaoyi Zhang, Richard Yuanzhe Pang, Clara Vania, Katharina Kann, and Samuel R. Bowman. Intermediate-task transfer learning with pretrained language models: When and why does it work? In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 5231–5247, Online, July 2020. Association for Computational Linguistics.

[9] Masaya Tsunokake and Shigeki Matsubara. Classification of urls citing research artifacts in scholarly documents based on distributed representations. In **2nd Workshop on Extraction and Evaluation of Knowledge Entities from Scientific Documents**, 2021.

[10] Yasa M. Baig, Alex X. Oesterling, Rui Xin, Haoyang Yu, Angikar Ghosal, Lesia Semenova, and Cynthia Rudin. Multitask learning for citation purpose classification. In **Proceedings of the 2nd Workshop on Scholarly Document Processing**, pp. 134–139, Online, June 2021. Association for Computational Linguistics.

[11] Iz Beltagy, Kyle Lo, and Arman Cohan. SciBERT: A pretrained language model for scientific text. In **Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing**, pp. 3615–3620, Hong Kong, China, November 2019. Association for Computational Linguistics.

[12] Alex Wang and Kyunghyun Cho. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In **Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation**, pp. 30–36, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[13] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, **Advances in Neural Information Processing Systems**, Vol. 31. Curran Associates, Inc., 2018.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. December 2014.

[15] He Zhao, Zhunchen Luo, Chong Feng, Anqing Zheng, and Xiaopeng Liu. A context-based framework for modeling the role and function of on-line resource citations in scientific literature. In **Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing**, pp. 5206–5215, Hong Kong, China, November 2019. Association for Computational Linguistics.