根拠に基づいたレビュー生成のための LLM を用いた自動アノテーションの検討

田中 翔平 [1] 平澤 寅庄 [1] 牛久 祥孝 [1]
[1] オムロンサイニックエックス株式会社
{shohei.tanaka, tosho.hirasawa, yoshitaka.ushiku}@sinicx.com

概要

本研究では論文の内容を根拠としてレビューを生成可能なモデルを構築することを目指す．この目的を達成する一つの方法として，学習データのレビューが論文のどの部分に基づいているかという情報を活用することが考えられるが，こうした情報を含むデータセットを人手で作成することはコストがかかる．そこで本論文では，レビューと論文の結びつきや論文の内容を正確に反映したレビュー文を LLM を用いて自動でアノテーションすることを検討した．評価結果より，現在の LLM はレビューと論文の結びつきをある程度の精度で予測することができ，論文の内容を正確に反映したレビュー文を高い精度で選択できることがわかった．

1 はじめに

国際会議やプレプリントサーバーに投稿される論文の数は年々増加しており，特に査読者にかかる負担の増加は研究コミュニティにとって重大な課題となっている [1]．この問題を解決する一つの手段として，LLM を使用して論文レビューを自動で生成することが考えられる．LLM を使用したレビュー生成は盛んに研究されているものの [2, 3]，多くの国際会議では LLM のみを使用して自動でレビューを生成することは禁止している [1)]．これは現在の LLM は人間のレビュアーと比較して対象論文の欠点について指摘することが難しく，また事実と異なるレビューを生成することがあるためである [2, 4]．しかし，LLM が人間のレビュアー以上に正確なレビューを生成できるようになれば，レビュアーの負担が軽減されるだけでなく，論文の著者も不正確なレビューに対応する手間が軽減されるため，研究サイクルがより円滑になると予想される．

LLM によるレビュー生成の性能を改善する方法として，レビュー対象の論文を明確な根拠としてレビューを生成することが考えられる．現在公開されているレビューデータセットはレビューデータ単体で完結しており，論文本文との結びつきについてはアノテーションされていないものが多い [2, 4]．NLPEER [5] では “Line 114: please elaborate...” など明確に論文の特定の箇所に言及している文のみレビュー文と論文の結びつきが自動でアノテーションされているが，レビュー全体についてはアノテーションされていない．このため論文の特定のセクションを明確な根拠としてレビューを生成するモデルを構築するためには，レビュー文が論文のどのセクションに基づいているかというアノテーションを含んだデータセットを構築する必要がある．

また現在のレビューデータセットは人間が記述したレビューをすべて学習データとして使用することが一般的であるが，実際には人間が記述したレビューの中にも論文の内容を正確に反映していないものが含まれている [4]．こうした質の低いレビューを学習データから取り除く方法として，メタレビューに同様の内容が記述されているレビュー文のみを学習データとして使用することが考えられる．メタレビュアーは通常レビュアーよりも経験豊富な研究者が務めることが多く，また複数のレビューを踏まえてメタレビューを記述するため，メタレビューにも同様の内容が含まれるレビュー文は論文の内容を正確に反映している可能性が高い．

こうした科学論文を対象としたアノテーションには高度な専門知識が必要であり，データセットを作成するコストが高いため，人手で大規模なデータセットを作成することが難しい．この問題に対処するため，本研究ではレビューと論文の結びつきや論文の内容を正確に反映したレビュー文を LLM を用いて自動でアノテーションすることを検討した．具

1） https://aclrollingreview.org/reviewerguidelines#q-can-i-use-generative-ai

体的には，レビューと論文の結びつきについてはLLMにレビュー文と論文本文のペアを与え，与えられたレビュー文が論文のどのセクションに基づいているかを予測させた．また論文の内容を正確に反映したレビュー文の選択については，LLMにレビューとメタレビューのペアを与え，メタレビューと同様の内容を含むレビュー文を選択させた．実験結果より，現在のLLMによるレビュー文と論文の結びつきの予測は約66%のaccuracyであり，このタスクについてLLMをある程度自動アノテーションに活用できることが判明した．一方，LLMによる正確なレビュー文の選択については90%以上のprecisionとなっており，このタスクについてはLLMを高い精度で自動アノテーションに活用できることが判明した．

2 関連研究

論文レビューを生成する初期の取り組みとしてはReviewAdvisor [6]やReviewRobot [7]が知られている．ReviewAdvisorはBART [8]ベースのモデルを用いて論文を要約し，要約に基づいてレビューを生成するという2段階のシステムを採用している．生成されたレビューの評価指標としては，実際のレビュースコアやレビューとの類似度の自動評価，人手によるレビューの質の主観評価が用いられた．評価の結果，ReviewAdvisorは事実と異なる内容を生成してしまう，学習データ中に含まれる文言を繰り返してしまう，といった欠点があることが判明し，ReviewAdvisorはあくまで人間によるレビュー生成の補助に使用するべきだと著者らは述べている．ReviewRobotはレビュー対象の論文や関連研究から知識グラフを構築し，その知識グラフに基づいてレビュースコアの予測とレビュー生成を行う．レビュースコアについては実際のレビュースコアとの類似度を自動評価し，レビューについては人間のレビュアーが主観評価を行うことでモデルの性能を評価している．その結果，ReviewRobotはレビュースコアについては高い精度で予測できているものの，レビューについてはテンプレートを用いて生成しているため，人間のレビューと比較して一般的すぎるコメントをしてしまうという問題を抱えていることが明らかになった．ReviewerGPT [9]はGPT-4などのLLMが論文レビューにどのように役立つかを調べるため，論文中に挿入した誤りの検出や優れた概要の選択といった実験を行ったものである．この実験ではLLMによる論文の評価について人間が個別に結果を確認しており，また大規模な評価は行っていない．Zhouら [2]はGPT-4などのLLMを使用してレビュースコアの予測やレビュー生成を行った．レビュースコアについては自動評価を行い，レビューについては自動評価と主観評価の両方を実施し，また実際のレビューと生成されたレビューの関連性，精度，再現率といった0-100スケールのスコアの評価にはGPT-4も使用している．評価結果より，GPT-4によるスコア評価は人間による評価との違いが大きく，GPT-4は詳細な批評を生成できないことが明らかになった．Yuら [3]はモデルが生成したレビューを自己改善するフレームワークSEAを提案した．この研究では人間のレビューとモデルが生成したレビューの類似度については自動評価を行い，またレビューの質についてはGPT-4を用いたスコア付けも行った結果，GPT-4は人間と比較してどのレビューに対しても高いスコアをつけることが明らかになった．Duら [4]はLLMが生成するレビューと人間が生成するレビューの違いを比較分析するため，生成されたレビューの欠陥を主観で評価した．またそうした欠陥をLLMが検出できるかについても評価した結果，LLMは人間と比較してレビューの欠陥を指摘する能力が低く，また本来不採択であった論文にも高いスコアを付けてしまうことが明らかになった．上記の先行研究にて提案されたモデルはすべて自動評価ではある程度の精度を達成しているものの，人間のレビューと比較した主観評価の結果，過度に肯定的なレビューや事実と異なるレビューを生成してしまうことが多いという共通の問題が明らかになっている．またGPT-4をレビューの評価に用いた実験では，GPT-4はレビューの欠陥を指摘することが難しいという問題が明らかになっている．すなわち，現在のLLMは論文のレビューとレビューの評価両方において根拠に基づいた適切な批判をすることが難しいという欠点を抱えている．

3 LLMによる論文レビューデータの自動アノテーション

本章ではレビューと論文の結びつきの予測，正確なレビュー文の選択という2つのアノテーションについて，LLMを用いて自動アノテーションを行うことを検討する．アノテーション対象のデータとしてSEA [3]に含まれるICLR2024のレビューデータから，採択された論文と不採択となった論文をそれ

ぞれ 5 本ずつ抽出して使用した．論文 PDF はあらかじめ Nougat [10] を用いて Markdown 形式のテキストファイルに変換されている．また論文のレビューデータには複数のレビューと 1 つのメタレビューが含まれており，JSON ファイルとしてパースされている．各レビューには Summary, Strengths, Rating といった記述式や評点式の複数の評価項目が含まれており，本研究では記述式の項目である Summary, Strengths, Weaknesses をアノテーション対象とした．また各レビューを Stanza [11] を用いて文に分割し，分割されたレビュー文を単位としてアノテーションを実施した．アノテーション対象のレビュー文は Summary が 159 文，Strengths が 161 文，Weaknesses が 312 文の合計 632 文である．自動アノテーションを行う LLM として GPT-4o [2], Gemini-1.5 [3] を使用した．GPT-4o は OpenAI 社が 2024 年 5 月にリリースしたモデルであり，Gemini-1.5 は Google 社が 2024 年 5 月にリリースしたモデルである．

3.1 レビューと論文の結びつきの予測

レビューと論文の結びつきのアノテーションを含んだデータセットを構築するため，レビューに含まれる各文が論文のどのセクションに基づいているかを LLM を用いて自動アノテーションした．具体的には LLM に対して論文テキストとレビュー文のペアを与え，与えられたレビュー文が論文のどのセクションに基づいているかを選択させた．ここで，論文の各セクションの役割は通常 Introduction, Method, Experiments のように大まかに分類できるが，実際のセクション名は “GAIA” のような具体的な手法名などが記されていることも多い．本研究ではこうしたセクション名の表記ゆれを吸収するため，アノテーション対象のセクション名をあらかじめ Introduction, Method, Experiments のような汎用的なセクション名に分類し，LLM には汎化したセクション名を選択するように指示した．なおレビュー文が “The paper is well written.” のように論文全体について述べている場合，Introduction を選択するように指示した．

LLM がセクションを選択した結果について，選択結果が正しいかを人手で評価した．評価結果を表 1 に示す．表 1 より，どちらの LLM も Summary のレビュー文についてのセクションの選択が最も accuracy が高く，Weaknesses についてのセクションの選択が最も accuracy が低いことがわかる．これは Summary は論文の内容について大まかにまとめたものであり，レビュー文が基づいているセクションを特定することが比較的容易であるのに対し，Weaknesses は論文に記述されていない実験の不足の指摘や著者の主張に対して反対する意見を述べることが多く，表面的な理解ではレビュー文が基づくセクションを特定することが難しいためだと考えられる．また全体的な accuracy は約 66%であり，このタスクについて現在の LLM をある程度自動アノテーションに活用できることが判明した．

表 1 レビュー文が論文のどのセクションに基づいているかの自動アノテーション結果 (Accuracy)

Model	Summary	Strengths	Weaknesses	All
GPT-4o	70.44	60.87	52.88	59.34
Gemini 1.5 Flash	76.10	70.81	57.37	65.51

3.2 正確なレビュー文の選択

レビューデータの中から論文の内容を正確に反映したレビュー文のみを抽出するため，メタレビューと同様の内容が記述されているレビュー文を LLM を用いて自動で選択した．具体的には LLM に対してメタレビューとレビューのペアを与え，与えられたレビューの中からメタレビューと同様の内容が記述されているレビュー文のみを選択させた．ここでレビュー文がメタレビューと同様の内容であるとは，論文の特定の側面についてレビュアーとメタレビュアーが同じ意見を持っているということを意味する．例えばメタレビューに “This paper proposes a novel approach to LoRA,” と記述されており，あるレビュー文が手法について詳しく記述している場合，このレビュー文は「論文が LoRA の新しい手法を提案している」という点でメタレビューと一致しているため，メタレビューと同様の内容を記述しているとみなされる．逆にメタレビューに “The reviewers mentioned that this is a combination of existing methods and lacks novelty, but I believe it is novel for the following reasons,” と記述されており，あるレビュー文に “This is a combination of existing methods and lacks novelty.” と記述されている場合，このレビュー文とメタレビューは手法の新規性について異なる意見を述べているため，同様の内容が記述されているとはみなされない．

LLM が論文の内容を正確に反映したレビュー文

2） https://openai.com/index/hello-gpt-4o/
3） https://blog.google/technology/ai/google-gemini-next-generation-model-february-2024/

を選択した結果について，選択結果が正しいかを人手で評価した．評価結果を表 2 に示す．表 2 より，GPT-4o の全体的な precision は 90%を超えており，GPT-4o が選択したレビュー文は高い確率で論文の内容を正確に反映していることがわかった．一方 Gemini 1.5 Flash の recall は約 47%であり，Gemini 1.5 Flash は GPT-4o と比較してより多くの正確なレビュー文を選択できていることがわかった．ただしこの自動アノテーションの目的が論文の内容を正確に反映したレビュー文のみを抽出することであることを鑑みると，precision を高く保つことの方が recall を向上させるよりも重要であるため，この自動アノテーションを実施する際は GPT-4o を用いて正確なレビュー文を抽出した方が良いと考えられる．またレビュー項目ごとの結果に着目すると，Summary や Weaknesses における precision が高く，Strengths における precision が低くなっていることがわかる．これは Summary や Weaknesses に書かれているレビュー文はメタレビューにも表面的に同様の内容が記述されていることが多いが，Strengths に書かれているレビュー文は提案手法や実験結果のポジティブな点についてメタレビューよりも詳細に書かれており，表面的に一致しないことが多いためだと推測される．

追加実験として，論文本文とレビューが与えられた際，メタレビューの参考になりそうな論文の内容を正確に反映したレビュー文を LLM が選択できるかを調査した．LLM がメタレビューなしでレビュー文の正確さを評価できれば，メタレビューが存在しない論文に対して LLM が生成したレビューを人間の評価者を介さずに評価できるようになり，LLM による自動レビュー生成の学習について自己改善ループを構築することができるようになる．評価結果を表 3 に示す．表 3 より，メタレビューなしで論文の内容を正確に反映したレビュー文を選択する場合の全体の precision は約 70%であり，メタレビューありで選択する場合と比較して precision は大幅に低下することが分かった．しかし Summary についての precision はどちらのモデルも 2 ポイント未満しか低下しておらず，依然高い precision を保っている．これは Summary は論文のレビュー結果について要約する項目であり，レビューとメタレビューで記述されている内容が大きく変わらないためだと考えられる．一方 Strengths や Weaknesses について precison が大幅に低下しているのは，これらの項目のどの内容をメタレビューに含めるかがメタレビュアーに依存するところが大きく，メタレビューを参照できない状況で正確にレビュー文を選択することが難しいためだと考えられる．すなわちメタレビューに同じ内容が記述されている場合にレビュー文が論文の内容を正確に反映していると定義した場合，現状の LLM に論文本文とレビューのみを与えて改善ループを構築することは難しいことがわかった．

表 2　論文の内容を正確に反映したレビュー文の自動アノテーション結果（メタレビューを参考に選択）

Model	Review	Precision	Recall	F1
GPT-4o	Summary	97.44	29.46	45.24
	Strengths	86.36	21.59	34.55
	Weaknesses	95.24	10.20	18.43
	All	93.90	18.64	31.11
Gemini 1.5 Flash	Summary	87.34	53.49	66.35
	Strengths	73.77	51.14	60.40
	Weaknesses	83.33	40.82	54.79
	All	82.20	46.97	59.78

表 3　論文の内容を正確に反映したレビュー文の自動アノテーション結果（メタレビューなしで選択）

Model	Review	Precision	Recall	F1
GPT-4o	Summary	95.45	16.28	27.81
	Strengths	53.85	15.91	24.56
	Weaknesses	65.71	23.47	34.59
	All	68.64	19.61	30.51
Gemini 1.5 Flash	Summary	86.67	40.31	55.03
	Strengths	63.51	53.41	58.02
	Weaknesses	64.91	37.76	47.74
	All	69.76	41.89	52.34

4　おわりに

本論文では論文の内容を根拠としてレビューを生成するモデルを構築するために，レビューと論文の結びつきを含んだデータセットを LLM による自動アノテーションで作成する方法を検討した．具体的には，レビュー文が論文のどのセクションに基づいているかの選択，論文の内容を正確に反映したレビュー文の選択というアノテーションを LLM を用いて行った．評価結果より，現在の LLM はレビュー文が論文のどのセクションに基づいているかをある程度の精度で選択することでき，論文の内容を正確に反映したレビュー文は高い精度で抽出できることがわかった．今後はレビュー文が基づいてるセクションの選択の精度を向上させるとともに，レビューと論文の結びつきを含んだデータセットを構築し，論文の内容を根拠としてレビューを生成する手法について検討していく．

謝辞

本研究は，JST【ムーンショット型研究開発事業】【JPMJMS2236】の支援を受けたものです.

参考文献

[1] Nihar B. Shah. Challenges, experiments, and computational solutions in peer review. **Commun. ACM**, Vol. 65, No. 6, p. 76–87, May 2022.

[2] Ruiyang Zhou, Lu Chen, and Kai Yu. Is LLM a reliable reviewer? a comprehensive evaluation of LLM on automatic paper reviewing tasks. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, **Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)**, pp. 9340–9351, Torino, Italia, May 2024. ELRA and ICCL.

[3] Jianxiang Yu, Zichen Ding, Jiaqi Tan, Kangyang Luo, Zhenmin Weng, Chenghua Gong, Long Zeng, RenJing Cui, Chengcheng Han, Qiushi Sun, Zhiyong Wu, Yunshi Lan, and Xiang Li. Automated peer reviewing in paper SEA: Standardization, evaluation, and analysis. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, **Findings of the Association for Computational Linguistics: EMNLP 2024**, pp. 10164–10184, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[4] Jiangshu Du, Yibo Wang, Wenting Zhao, Zhongfen Deng, Shuaiqi Liu, Renze Lou, Henry Peng Zou, Pranav Narayanan Venkit, Nan Zhang, Mukund Srinath, Haoran Ranran Zhang, Vipul Gupta, Yinghui Li, Tao Li, Fei Wang, Qin Liu, Tianlin Liu, Pengzhi Gao, Congying Xia, Chen Xing, Cheng Jiayang, Zhaowei Wang, Ying Su, Raj Sanjay Shah, Ruohao Guo, Jing Gu, Haoran Li, Kangda Wei, Zihao Wang, Lu Cheng, Surangika Ranathunga, Meng Fang, Jie Fu, Fei Liu, Ruihong Huang, Eduardo Blanco, Yixin Cao, Rui Zhang, Philip S. Yu, and Wenpeng Yin. LLMs assist NLP researchers: Critique paper (meta-)reviewing. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, **Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing**, pp. 5081–5099, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[5] Nils Dycke, Ilia Kuznetsov, and Iryna Gurevych. NLPeer: A unified resource for the computational study of peer review. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, **Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 5049–5073, Toronto, Canada, July 2023. Association for Computational Linguistics.

[6] Weizhe Yuan, Pengfei Liu, and Graham Neubig. Can we automate scientific reviewing?, 2021.

[7] Qingyun Wang, Qi Zeng, Lifu Huang, Kevin Knight, Heng Ji, and Nazneen Fatema Rajani. ReviewRobot: Explainable paper review generation based on knowledge synthesis. In Brian Davis, Yvette Graham, John Kelleher, and Yaji Sripada, editors, **Proceedings of the 13th International Conference on Natural Language Generation**, pp. 384–397, Dublin, Ireland, December 2020. Association for Computational Linguistics.

[8] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, pp. 7871–7880, Online, July 2020. Association for Computational Linguistics.

[9] Ryan Liu and Nihar B. Shah. Reviewergpt? an exploratory study on using large language models for paper reviewing, 2023.

[10] Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. Nougat: Neural optical understanding for academic documents, 2023.

[11] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A python natural language processing toolkit for many human languages. In Asli Celikyilmaz and Tsung-Hsien Wen, editors, **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations**, pp. 101–108, Online, July 2020. Association for Computational Linguistics.