文献理解のための人間の応答を利用したプロンプト最適化

今川涼平[1]　守山慧[2]　楊明哲[2]　馬場雪乃[2]
[1]筑波大学大学院理工情報生命学術院　[2]東京大学大学院総合文化研究科
s2220565@s.tsukuba.ac.jp
{kei-moriyama,mingzhe-yang,yukino-baba}@g.ecc.u-tokyo.ac.jp

概要

大規模言語モデルを利用した自然言語システムの性能は，プロンプトの設計に影響される．人間の試行錯誤に基づく従来のプロンプト設計に代わる方法としてプロンプト最適化が取り組まれている．既存のプロンプト最適化手法では，正解ラベル付きデータの利用を前提としている．代わりに，システムの出力に対する人間からのフィードバックを利用することが考えられる．本研究では，学術論文のタグ抽出システムの運用を想定し，プロンプト最適化において，人間から得られるフィードバックの有効性を調査する．実験の結果，フィードバックを利用した最適化により，再現率の向上が確認できた．一方，精度の悪化も確認され，フィードバック設計の再考の必要性などの課題も確認した．

1　はじめに

学術分野の細分化や，学術研究の活発化に伴った論文数の増加のため，学術研究における関連文献調査にはより多くの時間や労力が必要となっている．関連文献調査を支援するため，学術論文に対して，その内容等を表すタグを自動で付与するシステムの構築を考える．各論文に付与されたタグの比較などを行うことにより，類似した文献を見つけることが容易となる．通常，このようなシステムの構築では，正解ラベル付きデータを用いて，機械学習モデルの学習を行う．しかし，大規模言語モデルの利用により，モデル学習を行わずにシステムを実現することができる．

昨今，大規模言語モデルが，その利用の容易さからさまざまな場面で活用されている．大規模言語モデルの利用においては，タスクに応じたモデル学習の代わりに，プロンプトの適切な設計により，特定のタスクに適用することが可能であるが，人間による従来のプロンプト設計には時間や労力がかかるという課題がある．この課題に対処するために，大規模言語モデルを用いたプロンプトの最適化が取り組まれている．これは，人間の試行錯誤を必要とせず，大規模言語モデルの文章生成能力を利用して，より良いプロンプトの獲得を目指す取り組みである．既存のプロンプト最適化手法では，最適化の過程で使用する正解ラベル付き学習用データを準備する必要がある．

タグ抽出システムの運用においては，システム利用者からタグに対するフィードバックを収集することが可能である．最も単純な，タグの正誤に対するフィードバックは，前述の正解ラベル付きデータの作成と異なり，システムが提示するタグを人間に評価してもらうことにより行える．タグの抽出は，論文文章に基づいて行われる．このため，人間はシステムの抽出タグを評価する際，文章中でタグについて言及し，タグ抽出の根拠となっている箇所を探し，その妥当性を評価すると考えられる．したがって，システムによりタグの根拠となった論文中の文がタグと同時に提示されれば，人間がシステムのタグ抽出過程を理解するのを助け，より適切なタグの評価が可能となる．この根拠文の抜き出しは大規模言語モデルの利用により行える．また，タグの根拠文に対するフィードバックは，プロンプト最適化において，タグの追加情報として利用することができる．

本研究では，プロンプトの最適化における，人間からのフィードバックの有効性について調査する．タグとその根拠のペアに対するフィードバックを用いることで，通常の学習用データよりも小規模なデータを用いたプロンプト最適化を目指す．より具体的には，まず，学術論文のタグ抽出システムの運用を想定する．システム利用者からシステムの出力に対するフィードバックを収集し，これを利用してプロンプトの最適化を行う．

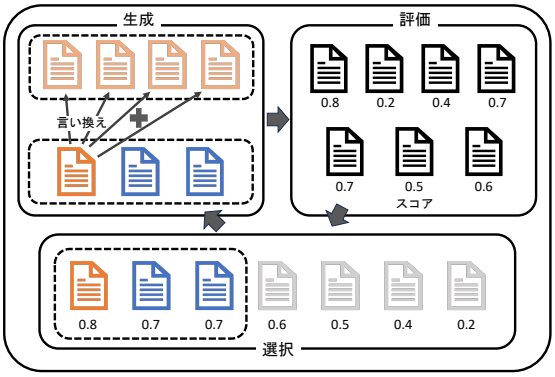

図 1: 本研究で採用するプロンプト最適化の流れ

2 関連研究

より良いプロンプトの獲得のため，プロンプトの言い換え [1] や，指示のランダムサンプリング [2] などが行われている．既存のプロンプト最適化手法の多くは，「生成」，「評価」，「選択」の繰り返しからなる [3, 4]．初期プロンプトを元に多様なプロンプトを生成し，正解ラベル付きデータを用いてこれらを評価し，評価結果に基づいて現在のプロンプトを取捨選択する．例として，進化アルゴリズム [5] や遺伝的アルゴリズム [6] を繰り返し適用する手法や，プロンプトを頂点，プロンプトの変更案を辺とする木の探索をおこなう手法 [7] などがある．これらの手法では，最適化の過程におけるプロンプトの評価に使用する正解ラベル付きデータを必要とするが，本研究では人間から得られるフィードバックを利用するため，事前のデータの準備の必要がないという違いがある．その他，新たな候補プロンプトの作成を指示する効果的なメタプロンプトの調査 [8] なども行われている．

3 提案手法

3.1 2 段階プロンプト

本研究では，タグ抽出システムとして，入力文の中からタグと関連のある箇所を抜き出す「抜粋」と，抜き出された文章からタグを抽出する「抽出」の，2 段階のタスクを行う．この 2 段階のタスクをモデルに明示的に行わせるため，「抜粋プロンプト」と「抽出プロンプト」の二つのプロンプトを設計する．抜粋プロンプトでは，タスクの説明と論文文章を与え，抜き出した文章のみをそのまま出力するように指示する．続く抽出プロンプトでは，抜粋プロンプトの適用により抜き出された文章を入力文とし，タスクの説明やタグの一覧を与えて，実際にタグを回答するように指示する．具体的なプロンプトの例を図 2，3 にそれぞれ示す．

3.2 人間からのフィードバックを用いたプロンプト最適化

3.2.1 フィードバックの利用

タグ抽出システムの利用者から取得する，タグに関するフィードバックを用いて，プロンプト最適化を行う．本研究で行う最適化の流れを図 1 に示す．採用するアルゴリズムは，繰り返しに基づく Zhou ら [3] の手法に概ね従うが，今回はタグ抽出のために二つで一組のプロンプトを採用しており，それぞれに対して最適化をおこなう．

最適化の流れとしては，まず初回のみ，あらかじめ準備した初期化用プロンプトで二つの候補プロンプトの集合をそれぞれ初期化する．各繰り返し回が始まると，現在の最良プロンプトを親として，新たな候補プロンプトを生成し，候補プロンプトの集合に追加する．続いて，各候補プロンプトを用いて大規模言語モデルをフィードバックデータに適用し，候補プロンプトを評価する．この評価の上位プロンプトを次の回に持ち越すように候補プロンプトの集合を更新する．新たな候補プロンプトの生成は，図 6 を用いて，大規模言語モデルによる最良プロンプトの言い換えにより行う．

3.2.2 フィードバックの取得

システムの改善のため，タグ抽出システムは，自らの出力に対してシステム利用者からフィードバックを収集する．フィードバックは，システムの出力に対するシステム利用者による評価のことである．具体的には，システムはタグとその根拠文をシステム利用者に提示する．提示された根拠文がタグの根拠として適切であるか否かをシステム利用者に回答してもらい，これを二値のフィードバックとして収集する．根拠文は，タグが抽出される論文文章から大規模言語モデルに抜き出させる．根拠文の抜き出しのためのプロンプトでは，対象のタグと論文文章を与え，タグについて言及しており，タグに最も関連の強い文を回答するように指示する．論文文章の複数の箇所がタグと関連していることも想定されるが，最有力な一箇所のみを抜き出させる．これは，

最も有力な根拠文がタグの根拠として適切でなければ，そのタグの抽出が適切でないと言えると考えられるからである．

前述のフィードバックを用いて，プロンプト最適化の中で候補プロンプトの評価を行う．評価方法は，2 段階のプロンプトそれぞれ別で準備した．一つ目のプロンプトは，予測タグに対して適切であると回答を得た根拠文を，プロンプトに対する出力に含むか否かの二値で評価する．二つ目のプロンプトは，フィードバックのタグを正解として，タグを正しく予測できるかを評価基準とする．

4 実験

4.1 実験概要

本実験では，学術文献からのタグ抽出に取り組む．学術論文として物体検知分野の論文を対象とし，各論文の提案手法と実験に関してタグを抽出する．提案モデルのバックボーンモデルや，利用されたデータ拡張手法に関する，3 カテゴリ，計 27 タグを事前に設定した．各論文について，適切なタグを全て抽出することをシステムの目標とする．最適化で獲得されたプロンプトの性能を，二つのベースライン手法と比較する．一つ目は，候補プロンプトの初期化に用いたプロンプトである．もう一つは，根拠文のフィードバックを用いた抜粋プロンプトの最適化の有効性を確認するための，タグのフィードバックを利用した抽出プロンプトの最適化のみを行い獲得されたプロンプトである．実験全体を通して，OpenAI[1)] の gpt-3.5-turbo-16k を利用する．タグ抽出性能を確認する評価指標として，再現率と精度を用いる．

4.2 データセット

実験で使用する論文は，Papers with Code[2)] の，COCO test-dev[3)] を用いた物体検知のリーダーボード[4)] に掲載されている中から，手法提案論文を 40 本選択した．arXiv[5)] からダウンロードした PDF ファイルのテキスト情報のみを使用する．要旨，序論，関連研究，参考文献の章と図には，事前に設定したタグに関する言及が含まれにくいため，タグ抽出の対象から除外する．対象論文 40 本のうち 25 本は評価用とし，残りの 15 本はフィードバックを収集するために用いる．評価用論文に対する正解タグは，専門家二人で同意を取りながら付与した．

4.3 フィードバックの取得

繰り返しの各回で新たに生成される各候補プロンプトの予測に対して都度フィードバックを取得するのはコストが大きい．このため，初期化用プロンプトを用いて抽出されたタグに対してフィードバックを収集し，最適化全体を通して使用する．フィードバックは二人の専門家から，合わせて 15 本の論文に対して，正例 106 個，負例 132 個の計 238 個を収集した．

4.4 プロンプト最適化設定

各回で生成する新たな候補プロンプトは 8 つとし，各回の評価結果上位 3 プロンプトを次回に持ち越すこととする．局所解回避のため，一つのプロンプトが親となれる上限回数は 2 回とし，上限到達後は次に良いものを親とする．最適化過程でのプロンプトの評価には，全フィードバックの中から各回でバッチとしてランダムに選択された 32 サンプルを使用し，それらに対する評価値の平均に基づいて候補プロンプトの順位付けを行う．今回収集したフィードバックの正例と負例の数の違いに対処するため，抽出プロンプトの評価に用いるフィードバックは，正例と負例が同数になるようにサンプリングする．プロンプト最適化の繰り返し回数は 8 回とする．

5 実験結果

5.1 フィードバックを用いた最適化で獲得されたプロンプトの性能

はじめに，本研究で取り組むタグ抽出における 2 段階のプロンプトの妥当性を確認しておく．抽出プロンプトのみの性能と，抜粋プロンプトも利用した 2 段階の性能を表 1 にそれぞれ示す．関連箇所の抜き出しを経ることにより精度の大きな向上が確認でき，2 段階プロンプトは本タスクにおいて妥当なプロンプト設計だと言える．

1) https://openai.com/
2) https://paperswithcode.com/
3) https://cocodataset.org/#home
4) https://paperswithcode.com/sota/object-detection-on-coco
5) https://arxiv.org/

表 1: プロンプト最適化の有無によるタグ抽出性能比較

手法	最適化対象		再現率	精度
	抜粋	抽出		
抽出タスクのみ		–	0.789	0.550
抜粋・抽出タスク	–	–	0.774	**0.604**
抽出のみの最適化	–	✓	0.764	0.591
抜粋・抽出の最適化	✓	✓	**0.843**	0.525

まず，人間からのフィードバックを用いたプロンプト最適化の有効性を確認する．プロンプト最適化を適用していないタグ抽出プロンプトと，それを原点としたプロンプト最適化により獲得されたプロンプトの性能をそれぞれ表 1 に示す．この結果から，プロンプト最適化を行ったことで，精度は悪化し，一方で再現率は向上していることが確認できる．続けて，根拠文のフィードバックを用いた抜粋プロンプトの最適化の有効性を確認するため，タグのフィードバックを利用した抽出プロンプトの最適化のみを行った結果を同じく表 1 に示す．この抽出プロンプトのみの最適化と比較して，抜粋，抽出プロンプトの両方を最適化した場合には再現率が大きく向上していることが確認できる．

5.2 最適化によるプロンプトの変化

プロンプト最適化により最終的に獲得されたプロンプトを図 4，5 に示す．これらを確認すると，最適化の最初の親プロンプトである図 2，3 から，語彙の選択，文の構成など，さまざまな変化がみられる．例えば，「combine」が「merge」に，「also」が「additionally」に置換されるといった単語レベル変化から，「don't answer “true” for ${category}s not explicitly mentioned」の「only mark ${category}s as “true” if they are explicitly mentioned」への言い換えのような表現の変化まで確認できる．一方，親プロンプトに含まれないような新たな指示の追加は確認されなかった．

6 考察

まず，プロンプト最適化の適用により初期プロンプトから再現率の向上が見られたが，これは抜粋プロンプト適用時の関連箇所の抜き出し漏れが減少したためと考えられる．フィードバックを用いた抜粋プロンプトの評価は，適切な根拠文がプロンプト適用時の出力に含まれるか否かで行った．この点で，根拠文に対するフィードバックを用いて，期待した最適化が行えているといえる．

一方で，フィードバックで適切とみなされた根拠文のみを利用するプロンプトの評価方法は，最適化の結果，精度が悪化した原因でもあると考えられる．これは，抜粋プロンプトの導入により精度の向上が見られたことから推察できる．また，実際，評価用論文から抜粋プロンプトで抜き出された文の，1 ページあたりの平均単語数は，最適化前でおよそ 180 単語であったが，最適化後では 357 単語とおよそ 2 倍に増加していることを確認しており，これは，適切な関連箇所の抜き出しを優先したプロンプト最適化により，タグと無関係な箇所の抜き出しが増えたことの現れだと考えられる．

今回のフィードバックの設計では，例えばある論文文章にシステムが与えた根拠文が，タグの根拠として不適切であると利用者にみなされた場合，仮に根拠として適切な箇所が他にあったとしてそのことを判定できない．この問題の解決としては，適切，不適切の二値のフィードバックだけでなく，正しい根拠文などのより詳細なフィードバックの設計，取得などが必要となる．

次に，抽出プロンプトのみの最適化でいずれのスコアも悪化がみられたが，この原因の一つは，適切とされた根拠文に対応するタグは適切な抽出であるという仮定であると考えられる．今回収集したフィードバックは，根拠文がタグの根拠として適切であるかどうかであった．そのため，例えば，大規模言語モデルによる根拠の抽出に誤りがあれば，適切であるタグが不適切とみなされてしまう．抽出プロンプトの最適化を適切に進めるためにも，フィードバックの設計には改善の余地があると言える．

7 おわりに

本研究では，学術文献のためのタグ抽出システムの運用を想定した実験を行い，プロンプト最適化におけるフィードバック利用の有効性の有無を確認した．システムに対する人間からのフィードバックを利用したプロンプト最適化により，再現性の向上が確認された．一方，精度の悪化が確認され，フィードバック設計の再考の必要性などの課題も確認した．

謝辞

本研究は，JST ムーンショット型研究開発事業 (JPMJMS2236-8) の支援を受けたものである．

参考文献

[1] Yihe Deng, Weitong Zhang, Zixiang Chen, and Quanquan Gu. Rephrase and respond: Let large language models ask better questions for themselves. **arXiv preprint arXiv:2311.04205**, 2023.

[2] Yao Lu, Jiayi Wang, Sebastian Riedel, and Pontus Stenetorp. Prompt optimisation with random sampling. **arXiv preprint arXiv:2311.09569**, 2023.

[3] Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers. In **Proceedings of the 2023 International Conference on Learning Representations**, 2023.

[4] Reid Pryzant, Dan Iter, Jerry Li, Yin Tat Lee, Chenguang Zhu, and Michael Zeng. Automatic prompt optimization with ”gradient descent” and beam search. In **Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing**, 2023.

[5] Qingyan Guo, Rui Wang, Junliang Guo, Bei Li, Kaitao Song, Xu Tan, Guoqing Liu, Jiang Bian, and Yujiu Yang. Connecting large language models with evolutionary algorithms yields powerful prompt optimizers. **arXiv preprint arXiv:2309.08532**, 2023.

[6] Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. Promptbreeder: Self-referential self-improvement via prompt evolution. **arXiv preprint arXiv:2309.16797**, 2023.

[7] Xinyuan Wang, Chenxi Li, Zhen Wang, Fan Bai, Haotian Luo, Jiayou Zhang, Nebojsa Jojic, Eric P Xing, and Zhiting Hu. Promptagent: Strategic planning with language models enables expert-level prompt optimization. **arXiv preprint arXiv:2310.16427**, 2023.

[8] Qinyuan Ye, Maxamed Axmed, Reid Pryzant, and Fereshte Khani. Prompt engineering a prompt engineer. **arXiv preprint arXiv:2311.05661**, 2023.

A 付録

```
From the following text from a paper in the field of
object detection, which proposes ${proposed_model},
please extract all sentences that mention ${
category_explanation}.
When you extract multiple sentences, please combine them
with "\n\n".
Please output only the combined sentences as is without
adding any additional explanations.
Please do not change any words from the text and do not
add anything like quotation marks.
If there is no appropriate part, just answer "null"
instead.

Text:
${text}
```

図 2: 抜粋プロンプト（関連箇所の抜き出し用）．${proposed_model}，${category_explanation}，${text} には提案手法名，抽出対象カテゴリの説明，論文文章がそれぞれ挿入される．

```
Please read the following text from a paper in the field
of object detection and select ${category_explanation}
from the following table for the proposed model ${
proposed_model}.
The output format should be in JSON, where keys are the
provided ${category}s in the table and values are either
"true" meaning selected or "false" meaning not selected.
Please include all the provided ${category}s as keys, and
 don't add any other ${category}.
Also, please don't answer "true" for ${category}s not
explicitly mentioned in the text.

${item_table}

Text:
${text}
```

図 3: 抽出プロンプト（タグの抽出用）．${proposed_model}，${category}，${category_explanation}，${item_table}，${text} には提案手法名，抽出対象カテゴリ名，抽出対象カテゴリの説明，タグ一覧，論文文章がそれぞれ挿入される．

```
Combine all sentences related to the ${
category_explanation} mentioned in the research paper
associated with the object detection and ${proposed_model
} by extracting and merging them using "\n\n". Output
only the merged sentences without any changes or
additional details. If no relevant section is discovered,
 respond with "null".

Text:
${text}
```

図 4: 最適化により獲得された抜粋プロンプト（関連箇所の抜き出し用）．${proposed_model}，${category_explanation}，${text} には提案手法名，抽出対象カテゴリの説明，論文文章がそれぞれ挿入される．

```
Please analyze the given table and choose the most
suitable ${category_explanation} from a paper about
object detection that accurately describes the proposed
model ${proposed_model}. Present the outcome in JSON
format, where each ${category} mentioned in the table
should be a key. Set the corresponding value as "true" if
 the ${category} is explicitly mentioned in the text. If
the ${category} is not mentioned, set the value as "false
". Make sure to include all the provided ${category}s as
keys in the JSON without adding any extra ${category}.
Additionally, only mark ${category}s as "true" if they
are explicitly mentioned in the text.

${item_table}

Text:
${text}
```

図 5: 最適化により獲得された抽出プロンプト（タグの抽出用）．${proposed_model}，${category}，${category_explanation}，${item_table}，${text} には提案手法名，抽出対象カテゴリ名，抽出対象カテゴリの説明，タグ一覧，論文文章がそれぞれ挿入される．

```
Please rephrase the following instruction without
changing its meaning. Note that "${*}" are placeholders,
so please include them as they are.

### Instruction

"""
${text}
"""

### Rephrased Instruction
```

図 6: プロンプト言い換え用プロンプト．${text} には親プロンプトが挿入される．