
МОДЕЛЬ ГЕНЕРАЦИИ ПРОМПТОВ ДЛЯ LLM

Матвей Крейнин

kreinin.mv@phystech.edu

Пётр Бабкин

babkin.pk@phystech.edu

Мария Никитина

nikitina.mariia@phystech.edu

Ирина Заборянская

akshiira@yandex.ru

9 декабря 2024 г.

АННОТАЦИЯ

В последние годы быстрое развитие больших языковых моделей (LLM) открыло большие возможности для автоматизации и улучшения задач обработки естественного языка, включая генерацию и редактирование контента. Однако, для получения желаемого ответа от языковой модели требуется тщательно подобрать подробный промпт, соответствующий задаче. В этой статье анализируются способы построения языковых моделей, специально адаптированных для генерации и улучшения промптов для LLM. Цель состоит в улучшении качества, точности и связности выходных данных LLM, что поможет повысить интерактивные возможности LLM в творческих и профессиональных условиях. Предлагаются методы создания универсального промпта/универсального набора промптов для различных наборов LLM. В статье предлагается модель-генератор на основе Microsoft Phi-2, тестируется качество работы нескольких LLM

на обычных промптах и улучшенных с помощью предложенной модели.

Ключевые слова: Prompt optimization · LLM · Hard prompts

Содержание

1 Введение	4
2 Предыдущие исследования	6
3 Постановка задачи	6
4 Вычислительный эксперимент	6
5 Заключение	6

1 Введение

Появление *больших языковых моделей (LLM)* значительно улучшило качество обработки естественного языка. Однако производительность LLM сильно зависит от качества созданного запроса (*prompt engineering*). Промпт-инженеры разрабатывают запросы, наблюдая за реакцией определённой LLM. С ростом числа LLM, каждая из которых обучена на разных источниках данных и, следовательно, имеет разные веса и чувствительность к тем или иным сочетаниям токенов, создание запросов становится все более трудоемкой задачей. Решение этой проблемы заключается в разработке автоматизированного фреймворка, способного предлагать оптимизированные промпты и адаптироваться к различным LLM. Предыдущие исследования в основном фокусировались на обучении эмбедингов, что было эффективно для более ранних и меньших языковых моделей. Однако современные LLM требуют связных текстовых запросов, адаптированных к их конкретным спецификам обучения. В данной статье предлагается методология для предобучения легкой модели генерации промптов, которая может быть быстро дообучена улучшать промпты для конкретной LLM.

Генерация промптов стала одним из самых популярных методов улучшения качества работы больших языковых моделей в решении разнообразных задач *обработки естественного языка (NLP)*. С появлением мощных предобученных языковых моделей, таких как GPT [6] или BERT [3], появилась необходимость быстро улучшить качество модели, несмотря на её большой вес. В отличие от традиционных методов дообучения, которые требуют дорогостоящих обновлений параметров языковой модели для каждой задачи, настройка промпта занимает куда меньше ресурсов. Основным интересом исследований в области генерации промптов связан с поиском оптимальных промптов, которые улучшают производительность LLM в широком спектре задач при условии ограниченного количества примеров для обучения.

Для решения данной проблемы было проведено множество исследований с использованием различных подходов. Например, в [5] используется вспомогательная модель для дообучения на *мягких промптах*, то есть эмбединге промпта с допущением, что пространство эмбедингов непрерывно. У таких методов есть несколько недостатков. Например, им требуется доступ к архитектуре LLM [2], что обычно невозможно при взаимодействии с LLM через API. Кроме того, мягкие промпты, полученные с помощью этих методологий, менее интерпретируемы на естественном языке [7], так как их отображение на естественный язык может сильно искажать текст. Поэтому со временем больше внимания стали уделять оптимизации *тяжёлых промптов*, то есть непосредственно запросам на естественном языке [4]. Для их обработки используются, например, обучение с подкреплением или механизмы обратной связи на основе LLM [8]. Тем не менее, эти алгоритмы всё ещё требуют доступ к архитектуре LLM и полагаются на ненаправленные стратегии поиска Монте-Карло в семантическом пространстве промптов.

Мощные модели, такие как GPT-3.5, проявляют нежелание обрабатывать бессмысленные токены и приоритизируют понятный человеку текст благодаря обучению на задачах, основанных на инструкциях [1]. Это подчеркивает важность создания связных и понятных для человека промптов. Предыдущие техники оптимизации промптов в основном фокусировались на оптимизации пространства эмбедингов и затем на поиске ближайшего слова из словаря. Эти методы работали хорошо до GPT-2, но показали плохое качество на современных LLM.

В данной статье предлагается подход, направленный на обучение легко-весной языковой модели-промптера на небольшом наборе данных, включающем промпты и оптимизированные промпты. Кроме того, чтобы обеспечить адаптивность к различным LLM, каждая из которых обучается по-разному и проявляет чувствительность к различным токенам словаря для одной и той же задачи, проводится дополнительное дообучение языковой модели, сопровождаемое добавлением линейного слоя, адаптированного к специфике

целевой LLM. Затем оценивается производительность модели промптера на нескольких LLM.

В качестве усложнённого вариант поиска промптов предлагается рассмотреть специфику матожидания оптимальных промптов для нескольких LLM.

2 Предыдущие исследования

3 Постановка задачи

4 Вычислительный эксперимент

5 Заключение

Список литературы

- [1] Valeriia Cherepanova and James Zou. Talking nonsense: Probing large language models' understanding of adversarial gibberish inputs, 2024.
- [2] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning, 2021.
- [3] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692, 2019.
- [4] Anat Kleiman Siddharth Swaroop Finale Doshi-Velez Weiwei Pan Luke Bailey, Gustaf Ahlritz. Soft prompting might be a bug, not a feature. 2023.
- [5] Guanghui Qin and Jason Eisner. Learning how to ask: Querying lms with mixtures of soft prompts, 2021.
- [6] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.
- [7] Weijia Shi, Xiaochuang Han, Hila Gonen, Ari Holtzman, Yulia Tsvetkov, and Luke Zettlemoyer. Toward human readable prompt tuning: Kubrick's the shining is a good movie, and a good prompt too?, 2022.
- [8] Tianjun Zhang, Xuezhi Wang, Denny Zhou, Dale Schuurmans, and Joseph E. Gonzalez. Tempera: Test-time prompting via reinforcement learning, 2022.