

**ИССЛЕДОВАНИЕ, АНАЛИЗ И ВЫЯВЛЕНИЕ РАЗЛИЧНЫХ ПРАВИЛ
ФОРМИРОВАНИЯ ПАТТЕРНОВ ПРОМПТОВ ДЛЯ БОЛЬШИХ ЯЗЫКОВЫХ МОДЕЛЕЙ**

Дворников А.С. (ИТМО), Архангельская Е.П. (ИТМО),
Научный руководитель – преподаватель, Аминов Н.С. (ИТМО)

Введение. При взаимодействии с большими языковыми моделями (Large Language Models, LLM) одним из ключевых факторов качества ответа является промпт: чем он точнее описывает проблему пользователя и чем лучше подобран под конкретную модель, тем качественнее будет ответ. В связи с такой зависимостью результата от входных данных, на текущий момент существует множество способов получить наилучший ответ от модели: начиная от простых правил написания запроса вручную и заканчивая сложными алгоритмами с множеством параметров, оптимизированными под любую модель.

Многие из современных решений являются условно-бесплатными интернет-сервисами. Например, сервис `promptperfect.jina.ai` [1], который умеет оптимизировать промпты 4 разными алгоритмами, на разных языках, для разных моделей (в том числе и генерирующих изображения). В ходе тестирования выяснилось, что данный сервис умеет работать с разными запросами: от повседневных общего характера до узкоспециализированных. Единственная проблема – это закрытость исходного кода и дороговизна API.

Помимо интернет сервисов, существуют фреймворки для улучшения промптов и дообучения моделей под конкретные нужды пользователя. Например, фреймворк The ProTeGi [2] от Microsoft, который может на основе предобученных данных задать модели правильный шаблон для улучшения промпта. Работает он по методу градиентного спуска: ищет слабые места промпта с помощью LLM и движется в обратную сторону от них, при этом собирая пулл наилучших промптов с помощью алгоритма “бандита” [3].

Также есть фреймворк AutoPrompt [4], который предлагает оптимизацию промптов с помощью процесса калибровки. В ходе оптимизации система генерирует синтетические данные для граничных случаев и оптимизирует промпт на основе созданного набора данных и ответов на них [5].

Основная часть. Цель исследования заключается в анализе описанных решений и выявлении требований к разрабатываемой методике формирования промптов в рамках сервиса рецензирования сложных документов.

Для анализа и тестирования каждого из решений применяется следующий алгоритм:

1. создание датасета - исходные промпты и эталонные ответы;
2. получение ответа от LLM на исходный промпт без его изменения;
3. модифицирование промпта с помощью какого-либо решения;
4. получение ответа от LLM на модифицированный промпт;
5. сравнение двух получившихся ответов и эталонного ответа.

Выделены четыре основные категории промптов, на которых будут тестироваться существующие решения:

1. общие промпты, которые представляют из себя обычные рутинные вопросы (например, «как приготовить куриный суп»);
2. творческие промпты, которые подразумевают придумывание чего-либо (например, «придумай идею для игры в жанре фантастика»);
3. логические промпты, которые подразумевают применение логики (например, «верно ли, что $2 + 2 = 5$ »);

4. узкоспециализированные промпты, которые подразумевают под собой применение знаний из узкой сферы деятельности (например, «объясни, как работает вертексный шейдер»).

При тестировании фреймворков и сервисов использовалась модель GPT-4o mini от OpenAI [6]. Сервис `promptperfect.jina.ai` хорош тем, что он позволяет выбрать в качестве входных данных конкретную модель, количество алгоритмов оптимизации, а также язык, на котором будет производиться оптимизация. Фреймворк The ProTeGi универсален: он умеет работать со всеми категориями промптов и со всеми моделями по причине того, что он использует метод градиентного спуска, который основан на параметрах самой модели. То есть модель сама ищет недостатки пользовательского промпта и после в зависимости от них улучшает промпт в обратную сторону. Фреймворк AutoPrompt умеет генерировать синтетический датасет на основе только пользовательского промпта, дополняя его граничными случаями для учёта всех ситуаций, что является его сильной стороной.

Описанные выше методики объединяются, и на их основе получается алгоритм, который на вход принимает пользовательский промпт и некоторые настройки модели, а на выходе выдаёт улучшенную версию промпта.

Выводы. В ходе данного исследования были проанализированы и протестированы существующие решения проблемы улучшения промптов для получения лучшего ответа от LLM. В итоге была получена общая табличка с результатами работы различных сервисов и фреймворков.

Также был написан алгоритм, который позволяет улучшать промпт в зависимости от множества факторов, тем самым тонко подстраиваться под конкретную LLM. В будущем планируется внедрить данный алгоритм в сервис рецензирования сложных документов.

Список использованных источников:

1. Prompt Perfect [Электронный ресурс]. - 2024. - URL: <https://promptperfect.jina.ai/interactive> (дата обращения: 24.12.2024).
2. The ProTeGi [Электронный ресурс]. - 2024. - URL: https://github.com/microsoft/LMOps/tree/main/prompt_optimization (дата обращения: 28.12.2024).
3. Pryzant, R., Iter, D., Li, J., Lee, Y.T., Zhu, C. and Zeng, M., 2023. Automatic prompt optimization with "gradient descent" and beam search. *arXiv preprint arXiv:2305.03495*.
4. AutoPrompt [Электронный ресурс]. - 2025. - URL: <https://github.com/Eladlev/AutoPrompt/tree/main> (дата обращения: 09.01.2025).
5. Levi, E., Brosh, E. and Friedmann, M., 2024. Intent-based Prompt Calibration: Enhancing prompt optimization with synthetic boundary cases. *arXiv preprint arXiv:2402.03099*.
6. OpenAI [Электронный ресурс]. - 2025. - URL: <https://openai.com> (дата обращения: 03.02.2025).

Автор _____ Дворников А. С.

Научный руководитель _____ Аминов Н. С.