Исследование проблемы адаптации языковых моделей к человеческим предпочтениям

Крашенинников Егор, гр. 422

Санкт-Петербургский государственный университет
Математико-механический факультет
Кафедра статистического моделирования

Научный руководитель: к.ф.-м.н., доцент Шпилев П.В.
Рецензент: к.ф.-м.н. Пепелышев А.Н.

Санкт-Петербург
2022г.

Введение

- **Большие языковые модели** становятся все более эффективными при решении задач обработки естественного языка
- Хотелось бы, чтобы они давали качественные ответы на вопросы, а не пытались пародировать текст, написанный человеком
- В этой работе предложен метод выравнивания цели обучения языковых моделей с человеческими предпочтениями

Моделирование языка

- Языковое моделирование — это задача оценивания распределения вероятностей следующего слова в тексте на основе всех предыдущих
- Обычно делается марковское предположение, что будущее слово зависит только от нескольких предыдущих
- n-граммная модель языка (x — слова, символы или токены):

$$p(x_1, x_2, ..., x_m) \approx \prod_{t=1}^{m} p(x_t | x_{t-1}, ..., x_{t-n+1})$$

Моделирование языка

Языковые модели могут быть параметризованы нейронными сетями и обучены с помощью ММП:

$$\theta^* = \text{argmax}_\theta \prod_{i=0}^{k} \prod_{t=0}^{m_i} p(x_{i,t}|x_{i,t-1}, ..., x_{i,t-n+1}, \theta),$$

- θ — параметры нейронной сети
- k — количество текстов в обучающем наборе данных
- m_i — количество токенов в данном (i-м) тексте
- n — количество предыдущих токенов, от которых зависит модель

Обозначения

- $\mathcal{Q}_{\text{vague}}$ — множество неконкретных вопросов
- $\mathcal{Q}_{\text{clear}}$ — множество конкретных вопросов
- $\mathcal{Q}_{\text{clarifying}}$ — множество уточняющих вопросов
- $\mathcal{C}$ — множество уточнений
- $\mathcal{A}$ — множество ответов
- Large Language Model (LLM) — большая языковая модель

Постановка задачи

- Пусть имеется LLM с настроенными параметрами и задача фактических ответов на неконкретные вопросы:

$$\mathcal{Q}_{\text{vague}} \to \mathcal{A}$$

- **Гипотеза:** качество генерируемых ответов на неконкретные вопросы улучшится, если дообучить LLM всегда генерировать один уточняющий вопрос перед тем, как давать финальный ответ

$$\mathcal{Q}_{\text{vague}} \to \mathcal{Q}_{\text{clarifying}} \to \mathcal{C} \to \mathcal{A}$$

- **Цель** — проверить, насколько взаимодействие с человеком улучшает качество генерируемых ответов

Метод

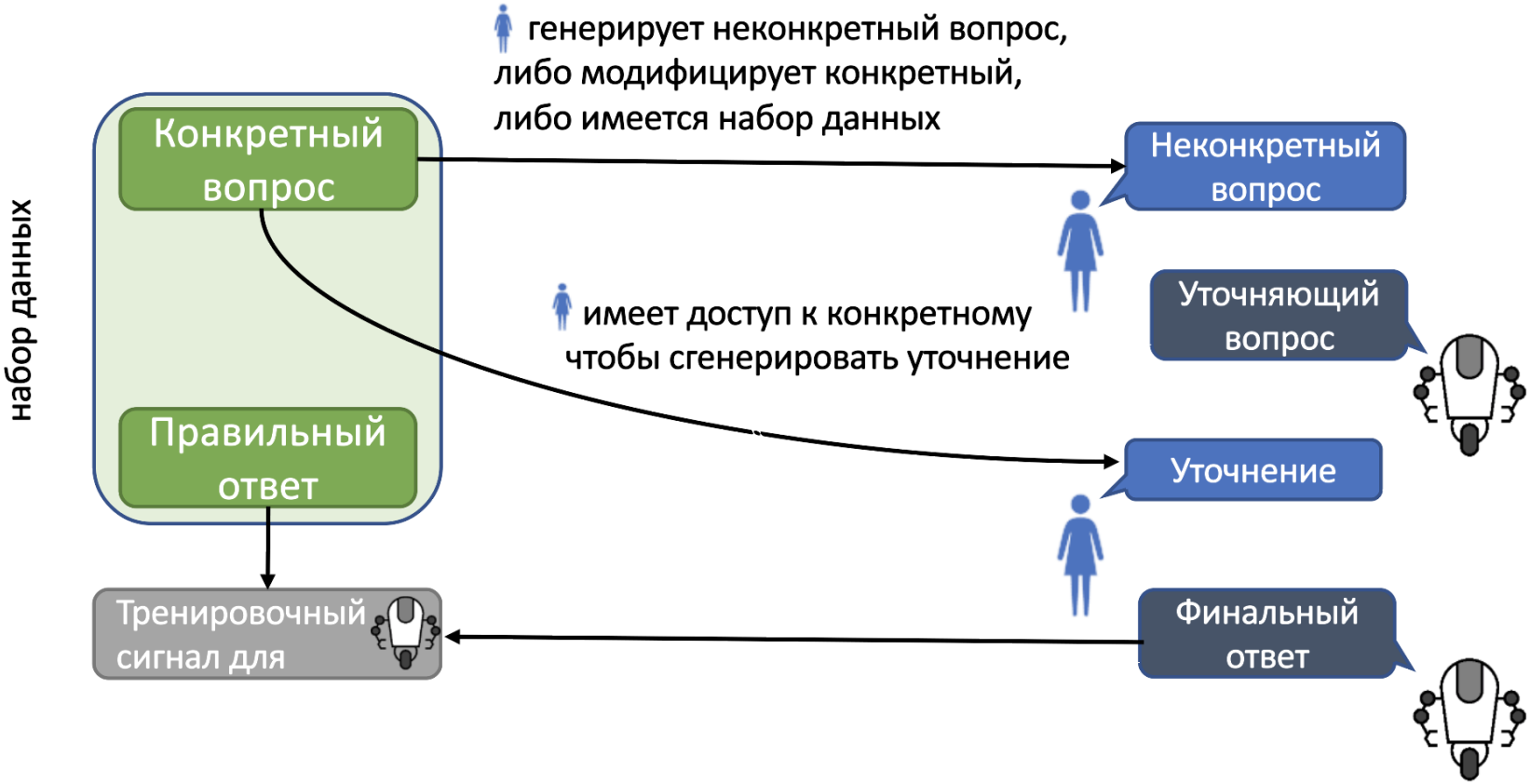

Рис.: Схема взаимодействия агента и человека

Метод

- LLM-агента можно представить двумя функциями:

$$A_q : \mathcal{Q}_{\text{vague}} \to \mathcal{Q}_{\text{clarifying}}$$

$$A_a : \mathcal{Q}_{\text{vague}} \times \mathcal{Q}_{\text{clarifying}} \times \mathcal{C} \to \mathcal{A}$$

- **Проблема:** реальные взаимодействия с человеком (уточнения) достаточно сложно получить

- Предлагается использовать еще одну LLM в качестве человеческого имитатора для генерации уточнений:

$$H : \mathcal{Q}_{\text{vague}} \times \mathcal{Q}_{\text{clear}} \times \mathcal{Q}_{\text{clarifying}} \to \mathcal{C}$$

Данные

- Набор данных с диалоговыми цепочками был составлен путем ручной разметки уточняющих вопросов и уточнений для примеров из AmbigQA (Min и др. 2020)
- AmbigQA $= \{q_{i_{\text{vague}}},\ q_{i_{\text{clear}}},\ a_i\}_{i=1}^{\approx 14k}$
- ClarifyingQA $= \{q_{i_{\text{vague}}},\ q_{i_{\text{clear}}},\ q_{i_{\text{clarifying}}},\ c_i,\ a_i\}_{i=1}^{\mathbf{1771}}$
- $q_{i_{\text{vague}}} \in \mathcal{Q}_{\text{vague}},\ q_{i_{\text{clear}}} \in \mathcal{Q}_{\text{clear}},\ q_{i_{\text{clarifying}}} \in \mathcal{Q}_{\text{clarifying}},\ c_i \in \mathcal{C},\ a_i \in \mathcal{A}$

Составление обучающих примеров H

Подбор правильной подсказки к входному тексту («prompt») — достаточно сложная задача. Для уточнения был подобран следующий «prompt»:

«prompt» для генерации уточнения

Clear question: <>
Vague question: <>
Clarifying question: <>
Clarification:

Составление обучающих примеров A

Для генерации уточняющего вопроса:

«prompt» для генерации уточняющего вопроса

Vague question: <>
Clarifying question:

Для генерации финального ответа:

«prompt» для генерации ответа

Vague question: <>
Clarifying question: <>
Clarification: <>
Answer:

Дообучение

- В качестве LLM использовалась модель GPT3 (Brown и др. 2020)
- A и H обучались **4** эпохи через OpenAI API
- **Базовый метод** — стандартный пример от OpenAI для QA, где входной текст шаблонизируется так:

«prompt» из стандартного примера QA

Q: <>
A:

- Им генерировались ответы для элементов множеств $\mathcal{Q}_{\text{vague}}$ и $\mathcal{Q}_{\text{clear}}$

Сравнение результатов

	davinci	**curie**	**babbage**
base-$\mathcal{Q}_{\text{vague}}$(базовый метод)	0.130	0.064	0.036
assistance (предложенный метод)	0.162	0.101	0.053
base-$\mathcal{Q}_{\text{clear}}$(максимум)	0.260	0.124	0.059

- Показано, что возможность задавать уточняющие вопросы человеку значительно улучшает качество в задаче генерации фактических ответов
- average F1 $= \frac{1}{n}\sum_{i=1}^{n} 2 \cdot \frac{P_i \cdot R_i}{P_i + R_i}$
- $P_i = \frac{|a_{i,\text{final}} \cap a_{i,\text{true}}|}{|a_{i,\text{final}}|}, \; R_i = \frac{|a_{i,\text{final}} \cap a_{i,\text{true}}|}{|a_{i,\text{true}}|}$
- $a_{i,\text{final}}$ и $a_{i,\text{true}}$ являются множествами слов соответствующей пары (i) ответов (предсказанный ответ и истинный)

Оценка релевантности уточняющего вопроса

- В предложенной схеме LLM-агент **всегда задает один** уточняющий вопрос
- Хотелось бы, чтобы он мог оценить, насколько релевантно задавать уточняющий вопрос и не задавать лишних
- Задание уточняющего вопроса уместно для всех $q_{\text{vague}} \in \mathcal{Q}_{\text{vague}}$ и неуместно для всех $q_{\text{clear}} \in \mathcal{Q}_{\text{clear}}$

$$\mathbf{X} = \{(q_i, r(q_i)) \mid q_i \in \mathcal{Q}_{\text{clear}} \cup \mathcal{Q}_{\text{vague}}\},$$

где $r(q) = \begin{cases} 1, & q \in \mathcal{Q}_{\text{vague}} \\ 0, & q \in \mathcal{Q}_{\text{clear}} \end{cases}$

Модели оценки релевантности

- Основной моделью для оценки релевантности была выбрана модель $\text{BERT}_{\text{base}}$ (Devlin и др. 2019)
  - Обучена на задачу маскированного языкового моделирования
  - Является композицией 12-ти кодировщиков из архитектуры Transformer (Vaswani и др. 2017)
- К ней был надстроен линейный слой с логистической функцией активации
- Иногда более простые классификаторы показывают похожие результаты, поэтому для сравнения взято несколько классических методов: метод ближайших соседей, логистическая регрессия, модель случайного леса, метод опорных векторов

Сравнение результатов

Model	ROC AUC	Accuracy	Precision	Recall	F1
LR	0.798	0.748	0.500	0.686	0.579
RF	0.822	0.815	0.730	0.425	0.526
BERT	**0.966**	**0.915**	**0.830**	**0.836**	**0.832**

Таблица: Значения метрик финальных моделей. LR — логистическая регрессия с l_1 регуляризацией ($\lambda = 2.0$), RF — модель случайного леса с энтропийным критерием. Все значения посчитаны с 5-folds кросс-валидацией.

Обобщенный метод

- Лучшая модель оценки релевантности (BERT) была использована для обобщения предложенного метода
- С вероятностью p_r, определяемой классификатором релевантности уточняющего вопроса, LLM-агент задает уточняющий вопрос, а с вероятностью $1 - p_r$ возвращает ответ базовой модели B_{vague}
- Базовые модели B_{vague} и B_{clear} были дообучены на всевозможных парах (q_{vague}, a) и на всевозможных парах (q_{clear}, a) соответственно (набора данных ClarifyingQA)

Обобщенный метод

- Большим недостатком является то, что модель агента представляют фактически **три** модели
- Чтобы не прибегать к использованию дополнительных моделей, предлагается составить данные для обучения агента, как описано ниже

«prompt» для генерации уточняющего вопроса (или ответа

Vague question: <>

«prompt» для генерации ответа

Vague question: <>
Clarifying question: <>
Clarification: <>
Answer:

Результаты

	davinci	**curie**	**babbage**
B_{vague}	0.157	0.100	0.052
assistance-generalized	0.202	0.141	0.071
B_{clear}	0.261	0.174	0.075

Таблица: Таблица значений average F1 между предсказанными ответами и действительными. «assistance-generalized» — агент, представленный тремя моделями.

	davinci	**curie**	**babbage**
B_{vague}	0.157	0.100	0.052
assistance-generalized-single	0.213	0.145	0.081
B_{clear}	0.261	0.174	0.075

Таблица: Таблица значений average F1 между предсказанными ответами и действительными для обобщенного агента, представленного **одной** моделью.

Заключение

- Показано, что возможность взаимодействия с (симулируемым) человеком действительно улучшает качество генерируемых ответов с помощью GPT3 на **неявные** вопросы из AmbigQA
- Предложен новый набор диалоговых данных — ClarifyingQA
- Успешно решена задача оценки релевантности задания уточняющего вопроса и лучшая модель использована для обобщения предложенного метода
- Реализован аналогичный обобщенный метод с использованием лишь одной модели GPT3, показавший схожие значения метрик