Министерство науки и высшего образования Российской Федерации
Федеральное Государственное бюджетное образовательное
учреждение высшего образования
«Московский авиационный институт»
(национальный исследовательский университет)

Институт № 8 «Компьютерные науки и прикладная математика»
Кафедра 805 «Математическая кибернетика»

Выпускная квалификационная работа на тему:

«Автоматическое формирование ответов в чате техподдержки с применением методов выравнивания языковых моделей»

Студент группы М80-405Б-20: Черных Сергей Дмитриевич

Руководитель: д.ф.-м.н., профессор, заведующий кафедрой 805
Алексейчук Андрей Сергеевич

Москва 2024

Постановка задачи

Разработать модель для автоматического формирования ответов на сообщения пользователей в чате технической поддержки сервиса. Модель будет реализована на языке Python с использованием библиотеки машинного обучения PyTorch, включая использование предобученных моделей. В работе будет использованы методы выравнивания языковых моделей для улучшения качества ответов. Будут построены несколько моделей, основанных на различных методах выравнивания языковых моделей, и их эффективность будет оценена, чтобы сделать выводы о наилучшем подходе к формированию ответов в чате технической поддержки.

Актуальность и новизна

Актуальность:

1. Высокая нагрузка на операторов чат-поддержки: чат-боты могут взять на себя часть рутинных задач, освободив время операторов для решения более сложных проблем.
2. Низкая скорость и качество обслуживания клиентов: чат-боты могут обеспечить круглосуточную доступность поддержки и более быстрый ответ на запросы клиентов.
3. Высокие затраты на обслуживание: чат-боты могут помочь снизить расходы на обслуживание клиентов.

Применение методов выравнивания языковых моделей:

1. Позволяет повысить точность и релевантность ответов чат-ботов.
2. Снижает риск предвзятости, дискриминации и других деструктивных действий.
3. Позволяет создавать чат-ботов, способных вести более сложные диалоги с клиентами.

Обоснование необходимости решения поставленной задачи

1. Снижение нагрузки на операторов чата
2. Повышение скорости и качества обслуживания клиентов
3. Круглосуточная доступность поддержки
4. Снижение затрат на обслуживание

Обзор известных методов решения поставленной задачи

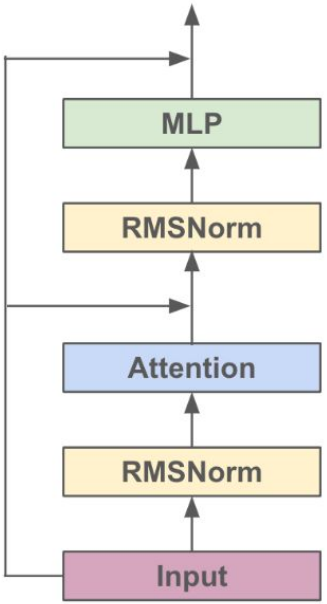
1. Sentence embeddings (векторных представлений предложений) для определения интентов пользователя в системах поддержки.
2. Prompt engineering: разработка подсказок, которые направляют генерацию текста в желаемом направлении.
3. Fine-tuning: выравнивание модели на небольшом наборе данных с целевой задачей.
4. Supervised fine-tuning: выравнивание модели на наборе данных с метками, указывающими на желаемые характеристики текста.
5. Reinforcement learning: использование обучения с подкреплением для обучения модели генерировать текст, соответствующий желаемым характеристикам.
6. Сбор данных, построение архитектуры модели, обучение модели.

Описание алгоритма

1) Выбор стека
2) Выбор метода промпт инжиниринга
3) Получение результатов на заготовленных данных
4) Оценка результатов

Модель для экспериментов

TinyLlama – компактная модель с 1,1 миллиардом параметров. Компактность позволяет обслуживать множество приложений, требующих ограниченного объема вычислений и памяти. Архитектура похожа LLama2, Однако было использовано меньшее количество экодер-блоков. Модель имеет высокие рейтинги по метрикам на HellaSwag, Obqa, WinoGrande, ARC_c, ARC_e, boolq piqa, в сравнении с аналогичными моделями с малым количеством параметров.



Эксперимент 1

1) Модель TinyLlama-1.1B
2) Фреймворк Pytorch
3) Промпт инжиниринг: простая генерация текста с дополнительным сообщением, что необходимо ответить.

Результаты:

Из-за малого количества ресурсов модель работала очень долго, не был сформирован корректный ответ. Модель генерировала диалог заказчика и сотрудника службы техподдержки сервиса.

Вывод:

Эксперимент оказался неудачным, нужны более продвинутые методы промпт инжиниринга.

Эксперимент 2

1) Модель TinyLlama-1.1B
2) Фреймворк Pytorch, Hugging Face
3) Промпт инжиниринг: персонализация и системные сообщения. Для модели подаётся системное сообщение, что она является ассистентом, заказчик ставится в роль user, модель в роль assistant.

Результат:

Улучшилось качество генерации, но значительно упала скорость, на доступных вычислительных мощностях генерация одного ответа модели занимала более 5 минут.

Вывод:

Эксперимент оказался неудачным. Слишком долгая генерация ответа.

Эксперимент 3

1) Модель TinyLlama-1.1B
2) Фреймворк Hugging Face, Langchain
3) Промпт инжиниринг: персонализация и системные сообщения. Для модели подаётся системное сообщение, что она является ассистентом, заказчик ставится в роль user, модель в роль assistant.

Результат:

Улучшилось качество генерации, чистота кода повысилась, простота интеграции в сервисы выросла, но значительно упала скорость, на доступных вычислительных мощностях генерация одного ответа модели занимала около 5 минут.

Вывод:

Эксперимент оказался неудачным. Слишком долгая генерация ответа.

Эксперимент 4

1) Модель TinyLlama-1.1B
2) Фреймворк Langchain, Ollama
3) Промпт инжиниринг: персонализация и системные сообщения. Для модели подаётся системное сообщение, что она является ассистентом, заказчик ставится в роль user, модель в роль assistant.

Результат:

Улучшилось качество генерации, чистота кода повысилась, простота интеграции в сервисы стала наибольшей из всех экспериментов, скорость генерации ответа составила в среднем ~ 13 с на один запрос на доступных вычислительных мощностях

Вывод:

Эксперимент оказался удачным. Лучшая скорость ответа, простота интеграции на наивысшем уровне сравнительно других фреймворков.

Оценка и интерпретация ответов

Ответы модели произведены по пятибалльной системе оценивания, каждый системный промпт рассматривается как отдельная модель, так как модель генерирует ответы на его основании.

1. "You are helpful assistant." – null
2. "You are a delivery tech support worker." – null
3. "Answer user complains as delivery tech support worker." – null
4. "You are helpful assistant. Answer user complains as delivery tech support worker." – null
5. "You are a delivery tech support worker. Answer user. Assure user everything will be solved." – null
6. "You are assistant at delivery service. Answer user. You get 1$ per good response." – null
7. "You are assistant at delivery service. Answer user. You get 10$ per good response." – null
8. "You are helpful assistant at delivery service. Answer user complains. You get 10$ per good response." – null
9. "You are helpful assistant at delivery service. Answer user complains. You get 100$ per good response." – null
10. "You are helpful assistant at delivery service. Answer user complains. You get 100$ per good response. Assure user everything will be solved and fixed" – null

Заключение

Для корректной работы модели в сервисе техподдержки необходимо:

1) Произвести Supervised fine-tuning для улучшения качества ответов модели, контроля этичности модели и избежания ответов на вопросы не касающиеся сервиса.
2) Произвести разметку оценок ответов модели асессорами, либо провести опросы в таргет группе сервиса.
3) Автоматизировать оценку ответов на основании разметки.

Выводы

1. Разработана модель автоматического формирования ответов в чате техподдержки, основанная на методах выравнивания языковых моделей.
2. Определены ограничения разработанной модели и пути её дальнейшего совершенствования.
3. Сформулированы рекомендации по внедрению разработанного метода в реальных условиях.

Литература

1. Борис Шапошников, Новые методы алаймента языковых моделей, Тинькофф, 2023
2. Никита Драгунов, Alignment языковых моделей. Prompt engineering & supervised fine-tuning // Practical ML Conf 2023, Яндекс Поиск, 2023
3. Ирина Степанюк, Интенты в саппорте на основе sentence embeddings и при чем тут LLM // Tinkoff.AI NLP Monolog Meetup #2, Тинькофф, 2023
4. Анатолий Потапов, Как собрать свой датасет для предобучения LLM // Tinkoff.AI NLP Monolog Meetup #2, Тинькофф, 2023
5. Tianhao Shen, Renren Jin, Yufei Huang, Chuang Liu Weilong Dong, Zishan Guo, Xinwei Wu, Yan Liu, Deyi Xiong. Large Language Model Alignment: A Survey, College of Intelligence and Computing, Tianjin University, 2023