Совместный хакатон НИУ ВШЭ – Нижний Новгород и Школы-21

Трек 1: Автоматический мониторинг качества модели через дашборд

Описание задачи

Проблема:

Мы хотим понимать, насколько хорошо наш чат-бот справляется с задачами. Для этого нужны метрики, которые будут автоматически собираться и визуализироваться.

Решение:

Создание сервиса, который:

- Собирает метрики качества работы модели (в т.ч. корректность, релевантность, точность контекста).
- Анализирует пользовательскую удовлетворённость (лайк/дизлайк).
- Отображает результаты в виде интерактивного дашборда (Streamlit/Dash).

Описание задачи

Элементы оценивания решения:

- Количество, обоснованность, понятность, адекватность и интерпретируемость предложенных метрик,
- Кастомизация и фильтрация метрик (по кампусам, по категориям вопросов, по вопросам, по пользователям, метрики производительности, и т.д., и т.п.),
- Визуализация и экспорт метрик по запросу.

Критерии успеха:

- Реализация дашборда для демонстрации метрик.
- Дашборд с визуализацией в реальном времени.
- (☆) Придумать свою метрику которая в онлайн режиме отслеживает некорректные ответы модели/галлюцинации. Реализовать.

Ресурсы и ограничения

Ресурсы:

- https://github.com/valerialevitskaya1204/hackathon_hse25.
- Исходный код базовых метрик находится в папке metrics/.

Ограничения:

- Фантазия и авторские права :)

Формат самбита

Презентация +

Модель: Для сабмита нужен пример как запускать Вашу модель.

Можно в виде функции, назовите ее inference, например.

Дашборд: Ваш дашборд в виде git-репозитория с README. Файл с необходимыми библиотеками (обязательно) и версиями (желательно). Пример как запускать.

Пожалуйста, протестируйте, прежде чем отдавать на проверку.

Если что-то не запустится при проверке, это автоматом понижение балла в 2 раза.

Мы, конечно, умеем дебажить, но давайте не забывать, что это конкурс ☺

Трек 2: Fine-Tuning модели для системного промпта

Описание задачи

Проблема: Для корректной работы нашей модели каждый раз необходимо передавать системный промпт. Это не всегда удобно, и мы хотели бы, чтобы модель изначально учитывала эти инструкции.

Решение: Провести fine-tuning модели, чтобы она автоматически следовала инструкциям, прописанным в системном промпте без дополнительных указаний.

Описание задачи

Элементы оценивания решения:

- Модель обрабатывает запросы в соответствии с системным промптом без его явной передачи.
- Качество ответов (остаётся на уровне или выше текущих метрик).

Ресурсы и ограничения

Ресурсы:

- https://github.com/valerialevitskaya1204/hackathon_hse25.
- Исходный код базовых метрик находится в папке metrics/.

Ограничения:

- Фантазия и авторские права :)

Формат самбита

Презентация +

Модель: Для сабмита нужен пример как запускать Вашу модель.

Можно в виде функции, назовите ее inference, например.

Дашборд: Ваш дашборд в виде git-репозитория с README. Файл с необходимыми библиотеками (обязательно) и версиями (желательно). Пример как запускать.

Пожалуйста, протестируйте, прежде чем отдавать на проверку.

Если что-то не запустится при проверке, это автоматом понижение балла в 2 раза.

Мы, конечно, умеем дебажить, но давайте не забывать, что это конкурс ☺

Немного полезных заметок

Немного про RAG

Для RAG есть скрипты и векторная БД которые лежат в папке

- https://github.com/valerialevitskaya1204/hackathon_hse25/rag_pipeline

Зачем вам эта информация?

1. Модель, что вы тюните – только часть всего пайплайна, а точнее ее конечный эндпоинт.
2. Перед эти должен запуститься RAG-pipeline, который найдет релеватные контексты.
3. Модель на основе этих контекстов должна сгенерировать ответ.
4. Проверяться качество вашей модели будет именно так, на каждый вопрос из тестовой выборки будет вызван RAG-pipeline.

Немного заметок (Валерия Левицкая, @vlone_l)

1. Если есть вопросы по RAG-pipeline, можете обращаться ко мне.

2. Если это поможет, можете пробовать менять сам RAG, но сразу предупрежу, что все параметры внутри были подобраны на основе экспериментов на реальных и синтетических сетах (т.е. их выбор был не случаен).

3. Не учите, пожалуйста, модели на много параметров. Модель обязана поместиться в общедоступные ГПУ на Кагле/Колабе. Если Ваша модель не поместится в ГПУ, она не будет принята.

Немного заметок (Валерия Левицкая, @vlone_l)

4. Системный промпт можно передавать при тестировании на вал-сете, во время тестов тоже будем подкладывать системный промпт, потому что понимаем, что на выборке из ~500 сложно обучить мощную модель, которая будет работать без сис-промпта.

5. Если Вы найдете лучший вид системного промпта (который будет показывать результат получше) можете предоставить и его, только сделайте отметку об этом.

6. С RAG то же самое – меняете пайплайн, указывайте это :)

7. По любым вопросам по модели/метрикам/промптам можно смело ко мне ☺

Немного заметок (Валерия Левицкая, @vlone_l)

8. Для тех кто делает задание с дашбордом – задание с метрикой (последнее)
стоит под звездочкой, но оно даст вам большое преимущество при оценивании работы,
так как для нашей команды это приоритет.

(Даже если вы не успели сделать конкретную реализацию,
накидайте хотя бы концептуально, уделите время на это задание)