

Как оценить результат обучения

Сначала определите полезный результат. Затем соберите свидетельства и проверьте одно изменение. Успех опыта — обоснованный вывод, включая отрицательный результат.

1. Создайте свою рабочую папку

Создайте папку **lecture02**, откройте её в Jupyter, создайте новый Python-блокнот и сохраните как **evaluation.ipynb**. Из **lecture02-adapter.zip** положите папку **pirate-adapter** непосредственно рядом с блокнотом. В ней нужны два файла: **adapter_config.json** и **adapter_model.safetensors**. Это сохранённый результат обучения; других файлов до первой ячейки не требуется.

Выберите подготовленное окружение: Python 3.11, torch 2.5.1+cu121, transformers 4.57.1, peft 0.17.1, sentencepiece 0.2.1, CUDA. Первые строки ячейки выбирают обычную HTTP-загрузку Hugging Face. Qwen скачивается кодом или берётся из кэша.

Полный блокнот со всеми ячейками приложен как справочник. На занятии создаём свой и переносим код со слайдов. Когда понадобится переводчик, на слайде 16 код сам скачает его в кэш: заранее класть модель в папку не нужно. На слайде 19 скачайте **cases.json** рядом с вашим блокнотом.

Полосы загрузки и строка об устройстве — обычные сообщения. Предупреждение о temperature, top_p и top_k означает, что параметры случайного выбора не участвуют при do_sample=False. Рекомендация sacremoses относится к необязательному нормализатору. Отдельные пояснения приведены на соответствующих слайдах. Traceback означает остановку ячейки и требует устранения причины.

Практика создаёт 17 ответов Qwen, включая начальный ответ и попытку перевода. Все исходные ответы сохраняются в **practice-results.json** рядом с блокнотом. Выполненная справочная версия позволяет читать опыт без запуска GPU.

2. Сформулируйте свою задачу

Кто будет использовать результат и что делать после ответа? Назовите полезный исход и критичную ошибку. Разделите содержание, выполнение инструкции, язык и стиль. Для точного числа или JSON стиль может быть неприменим.

Запишите наблюдаемые критерии до просмотра новых ответов. Для свободной задачи задайте обязательный смысл, недопустимые ошибки и примеры уровней оценки. Не требуйте одной эталонной формулировки там, где правильных формулировок много.

3. Подготовьте случаи и отложенную часть

Для самостоятельного упражнения начните минимум с 12 разработческих и 12 отложенных случаев. Включите несколько обычных задач, важные границы, неполные данные и ограничения формата. Это учебный старт, не доказательство достаточности набора для научного или промышленного вывода.

Сохраните идентификатор, запрос, группу, ожидаемое свойство и способ проверки. Проверьте повторы и близкие перефразировки с обучением. Отложенные случаи не используйте для выбора решения; после подбора по ним потребуется новый финальный набор.

4. Сравните версии при одинаковых условиях

Получите ответы исходной модели и модели с адаптером на одних запросах. Сохраните инструкцию, шаблон диалога, лимит и режим генерации. Сохраните все ответы, включая неудачные и оборванные. Отделите наблюдение от предположения о причине.

Если нужен перевод короткого английского ответа, используйте `result = T(answer).result.en` — оригинал, `result.ru` — перевод. Числа, JSON, язык и точные ограничения проверяйте по оригиналу. Уже русский ответ переводить не нужно.

5. Измерьте и объясните ошибки

Выбирайте автоматическое правило под конкретное требование. Указывайте, на каких случаях оно применимо. Доля всегда имеет числитель и знаменатель. Свободные ответы размечайте с пояснениями; непроверенные свойства не считайте успешными.

Покажите группы отдельно, если они отличаются назначением или важностью. Разберите как минимум один успех, одну ошибку и один спорный или частичный случай. Если используете помощника-модель для разметки, проверьте его решения на части ответов самостоятельно.

6. Проверьте одно изменение

Выберите повторяющуюся ошибку на разработческих случаях. Запишите гипотезу и измените один фактор: например, системную инструкцию. Это отдельный опыт, который не следует смешивать с оценкой самого дообучения.

Зафиксируйте решение и правила оценки, затем откройте отложенные вопросы. При ухудшении или частичном успехе сохраните результат как есть. Новая настройка по этой части уже требует другой независимой проверки.

7. Что сдавать

- Самостоятельный блокнот и короткие условия запуска.
- Сценарий, критерии, вопросы с назначением и группами.
- Модели, ревизии, адаптеры, настройки и все исходные ответы.
- Разметку с объяснениями и отдельные переводы, если использовали.
- Доли по применимым случаям, разбор ошибок, результат одного изменения.
- Вывод с ограничениями и предложением следующей проверки.

Форма вывода: «На ... случаях при ... условиях изменилось Это подтверждает ... в пределах Пока не проверено ...».

Готовый мини-опыт курса воспроизводит заранее известные автору случаи. Не выдавайте их за новый неизвестный тест своего решения. Дополнительный прежний прогон из 96 ответов — материал для разбора, а не ваши собственные новые генерации.

Помощь с запуском

Если адаптер не найден, проверьте, что `pirate-adapter` лежит непосредственно рядом с блокнотом и содержит оба файла. Для чтения вопросов `cases.json` должен находиться в той же папке. Модели Qwen и переводчика загружаются по их настоящим именам репозитория.

При отсутствии SentencePiece установите пакет в выбранное окружение, перезапустите ядро и выполните ячейки с начала. Если не хватает видеопамяти, завершите другие свои запуски моделей. Ошибки сети при первом скачивании нужно устранить; готовая справочная версия доступна для чтения.

Источники: [Qwen](#), [переводчик EN → RU](#), [pipeline](#).