**Планирование тестирования**

Чтобы облегчить тестирование системы на базе искусственного интеллекта, еще на этапе планирования процесса обратите внимание на следующие аспекты:

* **Попросите у заказчика как можно больше тестовых данных.**Чем больше у вас будет валидных входящих данных, тем проще составить адекватный data setup и начать обучение системе.
* **Соберите слаженную команду.**Может быть, это звучит банально, но в нашем случае этот аспект сыграл решающую роль. Учитывая, что системы на основе искусственного интеллекта самообучающиеся, при принятии командой неверных решений в работе с ней, легко можно научить ваш искусственный интеллект плохому. В результате программа будет мыслить неправильно и некорректно работать.
* **Спросите у Product Owner данные о проценте допустимой погрешности.**Так вы сэкономите себе много времени и нервов.
* **Учитесь думать нестандартно.**Не пытайтесь предусматривать действия искусственного интеллекта. Попытайтесь понять его нестандартную логику и тогда легко сможете взаимодействовать с ней.

**Подготовка к тестированию**

Прежде чем приступить к тестированию ПО с элементами искусственного интеллекта, убедитесь, что у вас:

* Налажен надежный контакт с заказчиком ПО. Только этот человек хорошо представляет, что именно он хочет получить от разрабатываемой системы.
* В команде есть специалист, ответственный за интеграцию. Он сможет как-то влиять на работу программного алгоритма.
* По возможности установите контакт с разработчиком ПО. Общение с ним поможет существенно облегчить понимание того, что написано в коде.
* Учите user flow. Если вы не будете досконально понимать, где, как и зачем ваш заказчик внедряет искусственный интеллект, вы не сможете понять, как именно его тестировать.

**Тестирование продукта**

Использование отрицательных тестов для самообученных алгоритмов нужно минимизировать, доведя их до одного-двух.

* **Помните: системы на основе искусственного интеллекта подходят для имитирования реального процесса принятия решений.**Поэтому при их тестировании отталкивайтесь от ситуаций, которые могут произойти с пользователем, когда он взаимодействует с продуктом.
* **Обеспечьте подключение постороннего инструмента ИИ с вашим приложением.**Выясните, как функционирует серверная часть и как происходит индексация данных. Это поможет избежать дефектов.
* **Постоянно тренируйте вашу систему.**Она же создана для того, чтобы учиться. Правильно подбирайте валидные тестовые данные исходя из данных production-сервера. Анализируйте результат не по сухим инструкциям по техдокументации, а по собственной логике. Имейте в виду, что отрицательные тесты следует использовать осторожно. Иначе можно научить систему «думать» неправильно.

Прежде чем передавать продукт в продакшн, убедитесь, что вы соблюдали следующие условия:

* Клиент знает, что продукты на основе искусственного интеллекта не могут работать «из коробки» идеально. Сначала их нужно научить, тренируя систему на валидных входных данных.
* Если у Product Owner нет таких данных, настаивайте на проведении бета-тестирования и UAT — пользовательского тестирования. Все же участие реальных людей в тестировании продукта не заменят никакие первоклассные тест-кейсы.
* Сразу тестируйте hot-fixes. Таким образом вы убедитесь, что разработчики правильно проанализировали найденные вами дефекты и внесли нужные корректировки в код. Но будьте готовы к тому, что тестирование может происходить в овертайме.