

Обзор отрасли

Михаил Трофимов, весна 2021

О чем поговорим

- Типичные задачи
 - С точки зрения теории
 - с точки зрения практики
- Research. Соревнования. Индустрия.
 - в чем между ними принципиальная разница
- Стадии проекта
 - Pre Sale, Pilot, Service Dev, Service Stable
- Роли в команде
 - DE / DS / MLE / MLO

Типичные задачи

С точки зрения теории и практики

С точки зрения теории

- 3 базовые задачи - классификация, регрессия, кластеризация
- Встречаются все три, но чаще всего именно классификация и регрессия
- Я бы добавил еще "генерацию" как отдельный случай, но это скорее редкость
- Самый частый случай - принятие решений - скорее классификация
- Будем в дальнейшем в голове держать именно случай классификации, но принципиальной разницы нет

С точки зрения практики

- Классические задачи
 - Рекомендательные системы
 - Поисковые системы
 - Churn prediction
 - LTV
 - Uplift modelling
 - Прогноз временных рядов
 - Антифрод (боты)
 - Антифрод (финансовый фрод)
 - Антифрод (запрещенный контент)

- Текст
 - Анализ тональности (чаще всего для отзывов)
- Картинки
 - Классификация картинок
 - Сегментация
 - Генерация
- Звук
 - Text-to-speech
 - Speech-to-text

Оно же, с точки зрения бизнеса

- Улучшать существующий автоматический процесс
 - Находить важные и срочные обращения в тех. поддержку с ML взамен поиска по ключевым словам
 - Улучшить ранжирование в поиске
 - Внедрить ML рекомендации взамен эвристик

- Улучшать существующий ручной процесс
 - Подсказывать тех. поддержке варианты ответа на обращение
 - Подсвечивать подозрительные участки на КТ

- Автоматизировать существующий ручной процесс
 - Автоматизировать клиентскую тех. поддержку
 - Self-driving cars
 - Автоматически проверять 90%+ объявлений на фрод перед публикацией

Это покрывает большую часть реальных задач, скорее всего над доведется поработать над чем-нибудь из этого списка.

Research. Соревнования. Индустрия.

Свойства ML-решения

Есть разные способы решения задачи, решение может иметь разные свойства, например:

- понятное (для заказчика)
- простое в поддержке
- быстрое (с точки зрения ресурсов при обучении)
- быстрое (с точки зрения ресурсов при инференсе)
- качественное (с точки зрения метрики)
- надежное/робастное (с точки зрения используемого подхода, фич и вероятности переобучения)
- научная новизна (самые современные модели, трендовость, неизвестные подходы)
- ...

Выделим три основных свойства/"оси": (научная) новизна,
качество, простота.

Почему они? Потому что ярче остальных отражают основные
подходы.

Новизна / Research

- Критерий - получился ли новый подход / модель.
- Заранее не известно, заведется ли
- Тут важно научиться решать новые задачи / новые подходы к известным задачам.
- В чистом виде - это академические ресерч.
 - иногда такое есть в крупных компаниях или технологических стартапах

Качество / соревнования

- Критерий - важна только метрика, пусть даже в ущерб всему остальному
- Больше фичей, больше разных моделей, больше стэкинга - это все сюда
- Потолок обычно не известен
- В чистом виде - соревнования
- Как правило, индустриальный рисерч - он именно про это.
 - Про перебор разных комбинаций в рамках задачи.
 - Ближе к соревнованиям, чем а академическом рисерчу

Простота / индустрия

- Критерий - модель простая в использовании
- Часто это компромиссный вариант между качеством и удобством в понимании/дебаге/поддержке
- Наиболее предпочтительный вариант для прода, особенно если моделей много
- Задача MLE - уменьшать техническую сложность и повышать надежность, чтобы как можно больше моделей попадали в эту категорию.

Примеры

Банковский скоринг

- Требуется интерпретируемость
- Робастность/надежность важнее качества

Трейдинг

- Качество на первом месте
- Никаких ограничений на ресурсы обучения
- Ограничение на время применения

Технологический стартап

- Качество на первом месте
- Задача может быть плохо изучена
- Деплоить/обслуживать толком нечего
- Нет ограничений по ресурсам

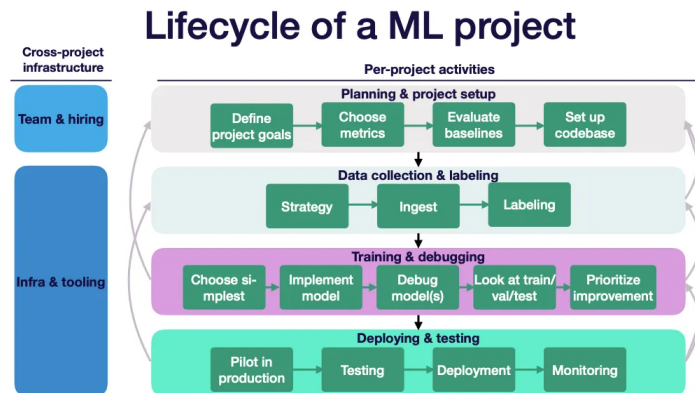
E-Commerce

- Моделей много, все надо обслуживать
- Есть продакшн, его ломать нельзя
- Мир меняется, продукт развивается, модели быстро устаревают
- Лучше брать модели на 2% хуже, но в 10 раз быстрее/легче в обслуживании

Стадии проекта

У проекта как бизнес сущности есть несколько условных стадий.

На разных стадиях - разные требования. Важно понимать текущую стадию проекта и его особенности.



<https://course.fullstackdeeplearning.com>

Стадии

- Pre-Sale
- Pilot
- Service Dev
- Service Stable

Pre-Sale

- Этап оценки технических требований и имеющихся данных
- Попытка найти нишу и возможность сделать продукт
- Часто “теоретический”

Pilot

- Технический прототип
- Должен работать, но не идеально
- Основная цель - доказать жизнеспособность идеи
- Как правило, нет жестких требований к скорости, качеству или надежности
- На этом этапе важна скорость разработки

Service Dev

- На этом этапе наращивается стабильность и качество сервиса
- Оптимизируется качество моделей, перфоманс
- Появляются требования к скорости, качеству, надежности

Service Stable

- Качество моделей уже почти не растёт
- Появляются требования к стабильности
- Логи, мониторинги, тесты

Роли в командах

Рассматриваем только технические. Некоторые роли могут отсутствовать, часто один человек совмещает несколько ролей (и не факт, что это хорошо)

- Data Engineer
- Data Scientist
- ML Engineer
- ML Ops

Data Engineer

- Делает так, чтобы данные стекались в нужное место в нужном формате
- Следит за наполнением/схемами/миграцией данных
- Следит за технической стороной работы хранилища, помогает профилировать тяжелые запросы.
- Как правило, пошарен между многими проектами
- Функция саппорта хранилища

Data Scientist

- Смотрит в данные
- Формулирует задачи
- Собирает датасеты
- Обучает модельки
- Центр бизнес-логики проекта

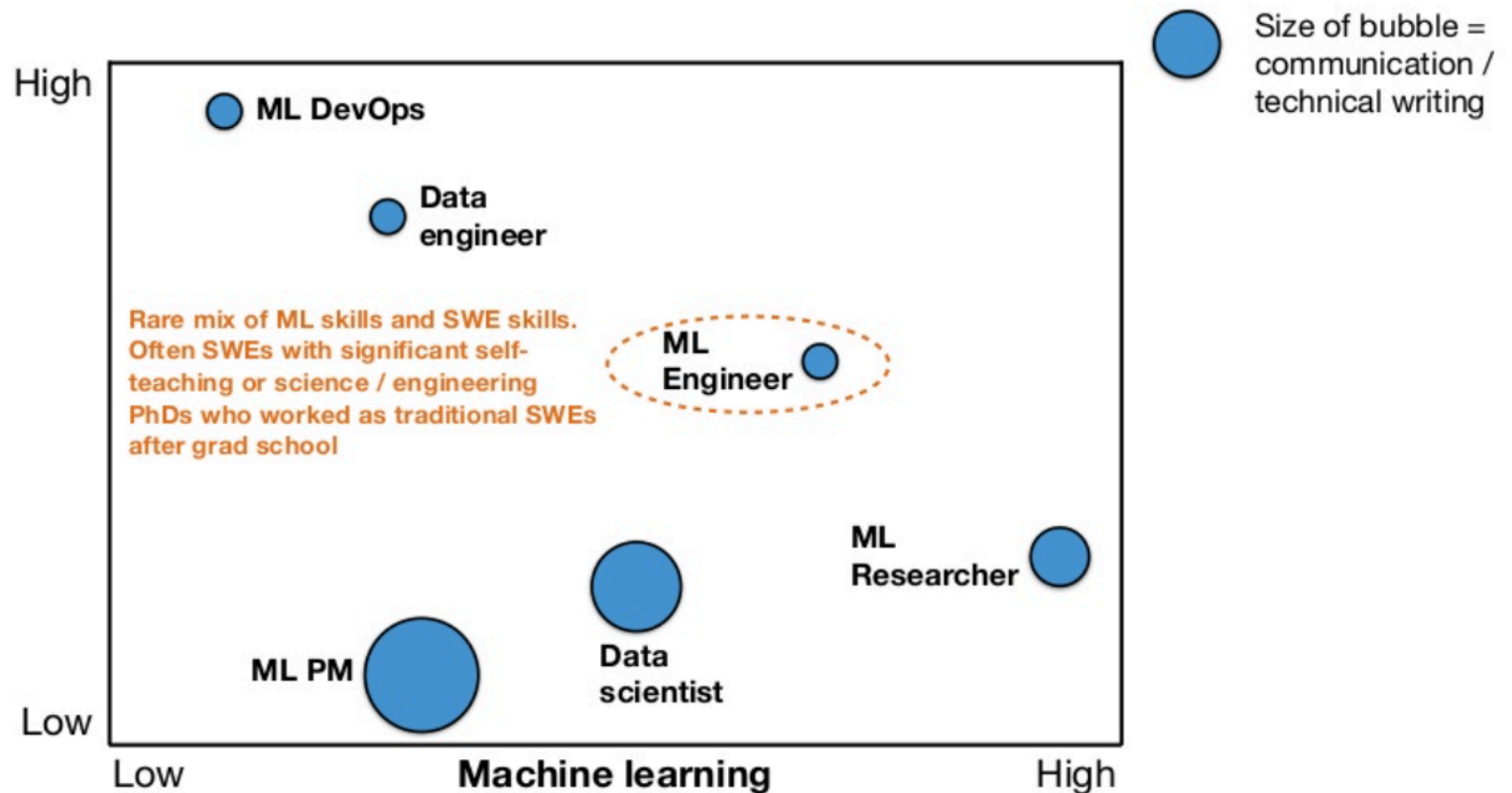
ML Dev / ML Engineer

- Решает технические проблемы интеграции / перфоманса.
- Превращает модельки в сервис / интегрирует с продуктом.
- Пишет тесты, ищет технические баги.
- Доводит модели до прода.

ML Ops

- Помогает выкатывать модельки
- Помогает строить пайплайны
- Настраивает логи/мониторинги/алерты
- Автоматизирует процессы
- Дает инструменты. Уменьшает сложность. Обслуживает продукт.

What skills are needed for the roles?



<https://fall2019.fullstackdeeplearning.com/course-content/ml-teams> (~17 слайд)

Примеры

В больших проектах как правило есть отдельно Data Engineer. Роль DS и ML Dev совмещена - то есть человек занимается постановкой задач, их решением и интеграцией решения в продукт. Роль MLOps берут на себя существующие в компании DevOps.

В технологических стартапах роль Data Engineer оказывается не востребоваанной, как и ML Ops. Основной движущей силой является DS и ML Dev, роль которого может выполнять, например, iOS Dev.

