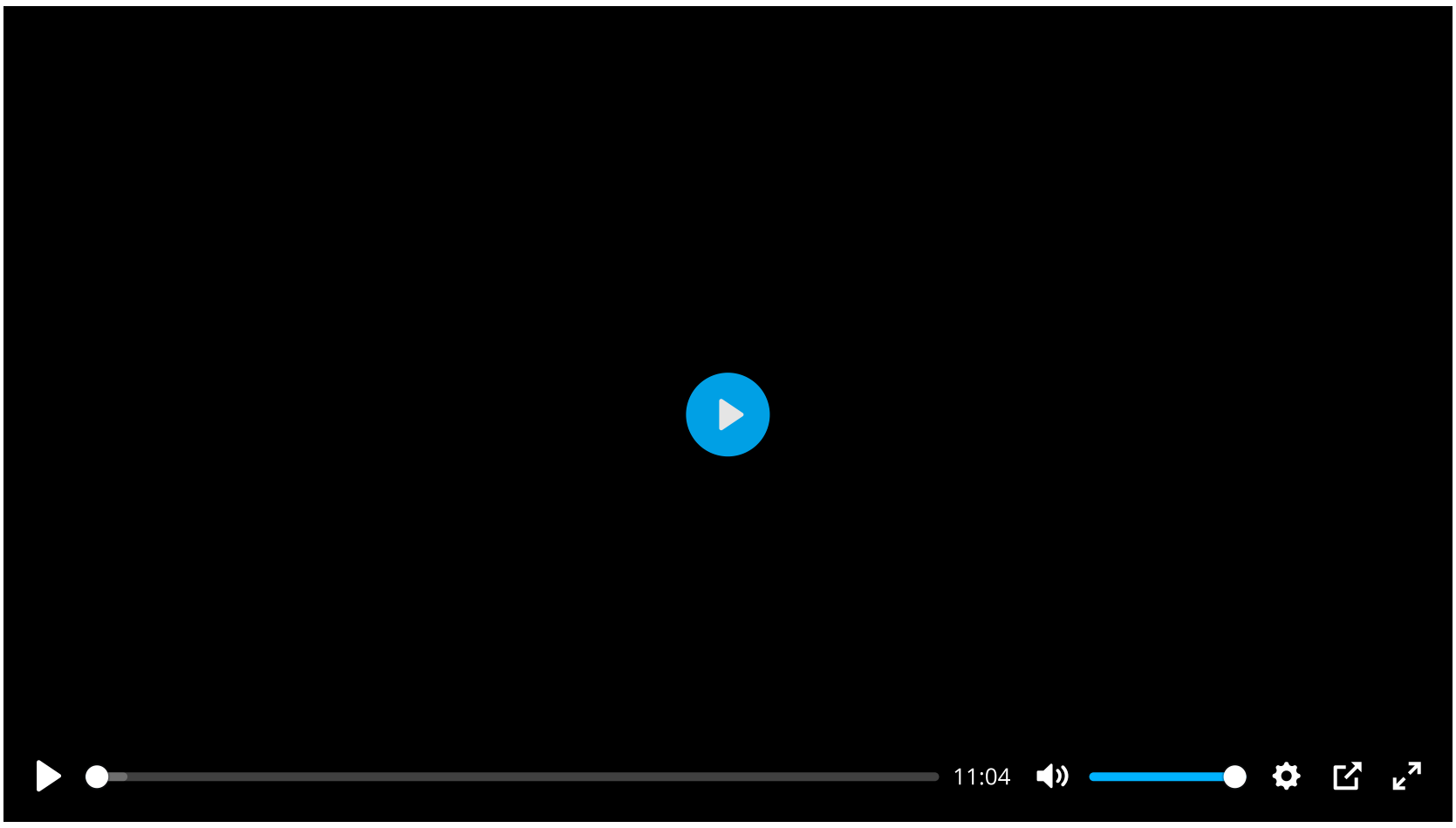

Текстовая расшифровка видео:

MODEL & FEATURE SERVING

План:

- Model Serving;
- ChatOps;
- Feature Store.

Model Serving

Концепция «Model Serving» появилась относительно недавно.

Инструменты:

- ML Flow Serving;
- TorchServe;
- Tensorflow Serving;
- NVIDIA Triton;
- BentoML.

Каждый фреймворк изобретает свой способ сервить модели.

На сегодняшний день один из самых частых вариантов развертывания – развертывание на платформе Kubernetes в докер-контейнерах. Там есть готовые механизмы для того, чтобы разворачивать все в облаке. Помимо этого, есть популярные сервисы – KServe, Seldon Core.

Суть сервисов проста: вы обучаете модельки, разворачиваете их, частично автоматизируете создание back-end, чтобы впоследствии сервить эти модели для инференса.

Не обязательно в качестве протокола для взаимодействия использовать «http», можно использовать «Jps».

Также существует «BentoML». Для него есть расширение, которое называется «Yatai». Это расширение позволяет модели сервить не только Standalone, но и Kubernetes, а также автоматизировать масштабирование нагрузки и т.д.

ChatOps

ChatOps помогает в организации взаимодействия с моделями.

У Github есть собственный бот, который называется «HU-BOT».

«HU-BOT» написан на языке программирования «Node.js». Бот имеет ряд ограничений.

Если вы хотите в своей компании внедрить возможность многофункционального бота, то обратите внимание на следующие проекты:

- **Opsdroid;**
- **Errbot.**

Данные проекты – реализации ChatOps. Это боты, умеющие работать с разными мессенджерами.

Feature Store

Вопрос: как люди, которые обучают модели, получают данные?

Ответ: для этого придумали **Feature Store.**

Модели машинного обучения используют наборы признаков в двух процессах:

- Обучение;
- Инференс.

Это может происходить как при исследовании, так и в продакшене.

Инициатором сбора файла может быть как Data Scientist, экспериментирующий с чем-то локальным, так и продуктовая модель, которая запрашивает данные.

Когда фич и моделей много, процесс получения выборок по конкретным фичам следует автоматизировать и регламентировать.

Есть готовые фреймворки, на которые стоит обратить внимание:

- **Feast** (по умолчанию рекомендуется к использованию On-Premise, позволяющей интегрироваться с разными источниками);
- **Tecton** (чуть более продвинутый фреймворк, но облачный).

Если ваша архитектура развернута на Amazon, рекомендуем присмотреться к Tecton.