Questions

Question - Rohit

Why is feature store connected to inference pipeline it can directly show predictions?

The feature store is the central repository of the features your ML models needs to generate predictions.

The income request coming from a client app has no features. These features are in the feature store. So, our REST API needs to connect to the feature store to fetch the freshest features, generate a prediction, and return it to the client.

Question - Emanuel

how many features pipeline can be built? as a best practice?

You start by building one, and see how far it gets you in terms of model evaluation metrics.

If it is enough, no need to build more.

If it is not enough, you can think of other data sources that make sense to ingest, so you build a feature pipeline, save the features in the store, re-train the model and check again evaluation metrics.

And so on…

Question - Gustavo

Are the features in a real-time ML system fixed as any other static ML model? i.e. in the housing price classic model you have size, location, year, etc, those are already fixed features to predict houses price

Yes, they are fixed. Every feature pipeline generates a predefined set of features.

Question - Alexander Openstone

Why is the feature store separate from the model? are you saying features fed

The feature store is a dedicated service for storing and serving ML model features, both at training time and at inference time.

The model is just a file that contains the function that maps ML models features (inputs) to predictions (outputs).

The model is generated by our training pipeline (which we will see next week) and saved to the model registry. From there, it can be loaded by the inference pipeline to generate predictions on new data.

Question - Jeofrey

Do we do EDA at he feature pipeline

No. You do EDA as part of your training pipeline.

Question - Alexander Openstone

Why do you need to recompute features per each request

To get the most relevant predictions I want to use the latest market data available. This is why our inference pipeline looks up the latest features from the store, feeds them to the model and generates the most up-to-date prediction.

Now, this simple workflow can be optimized with a cache. For example, if I got a request from a client and generated a prediction, I can re-use this exact prediction (no need to look up features in the store) for a new request entering one second later.

Question - Joshua Le

I really hope to learn a lot about building various feature pipelines (live + batch). Often In my work, the Data engineering part to build them is the bottleneck.

In this course we will build one real-time feature pipeline. If you want to learn batch pipelines, you can check my other course, the Real-World ML Tutorial.

A small question, will you cover environment segregation in future lessons? (DEV-STAGING-PROD)

Not sure if we will have time. I will try my best to get there :-)

Question - Adrian Guerra

What’s a topic

A Kafka topic is like a category or channel where messages are sent and stored. It's a way to organize data, so producers can send messages to a specific topic, and consumers can read from it. Think of it like a mailbox where multiple people can send and receive messages.

Question - Emanuel Tanasa

Can an AI system using RAG be set as a feature pipeline ?

For example: having a RAG system which collects financial news. Is it possible to set it as a feature pipeline?

I suppose if you use the architecture presented by you in all your system it is a plug in and that's it

Behind every RAG system there is at leat one program that is

> ingesting raw data (e.g. financial news),
> processing it (e.g. chunking and creating embedding representations), and
> storing it in a vector db.

This is a feature pipeline :-)