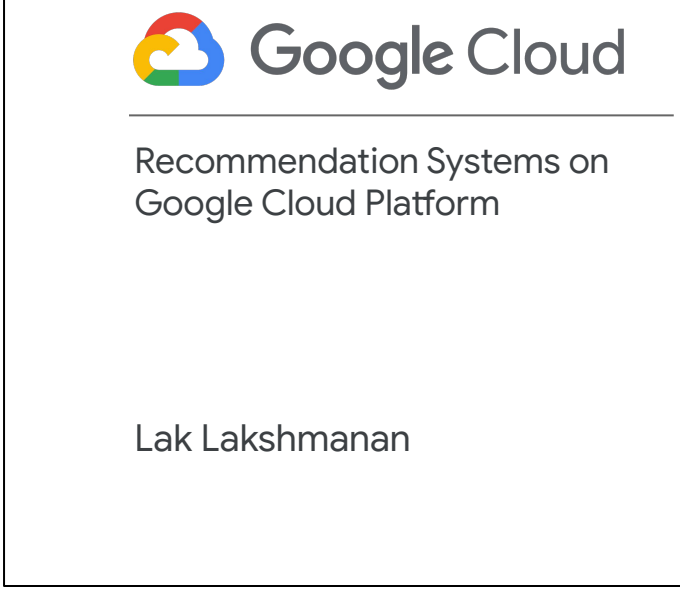


Hello, and welcome to our course on building recommendation systems with tensorflow on Google Cloud Platform.

My name is Lak, and I lead the team that is putting together this course and this specialization.

Specialization

~~End-to-End Lab on Structured Data~~

Production ML Systems

Image Classification Models

Sequence Models

Recommendation Systems

This is the fifth and final course of the Advanced Machine Learning on GCP specialization.

In this course, we'll learn how to build recommendation systems.
As usual, you will get hands-on practice building machine learning models on a variety of datasets in the labs we'll work on together.

Agenda

Introduction

Recommendation Systems

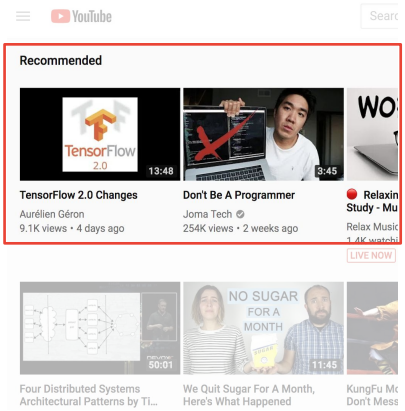
Content-based Recommendation Systems

Collaborative Filtering

Neural Networks for Recommendation Systems

Building an End-to-End Recommendation System

We start out by explaining what recommendation systems are.



When you watch a video on YouTube and you see a list of suggested videos to watch next, that list is being built by a recommendation machine learning model, often called a recommendation engine.

Agenda

Introduction

Recommendation Systems

Content-based Recommendation Systems

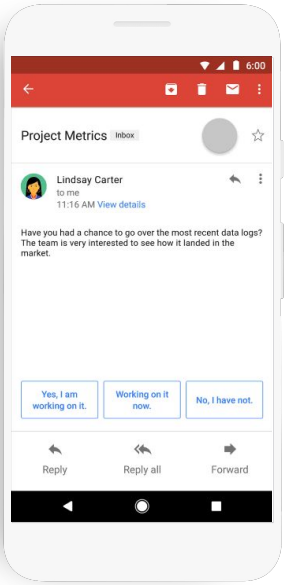
Collaborative Filtering

Neural Networks for Recommendation Systems

Building an End-to-End Recommendation System

Of course, just the ML model, just the recommendation engine is not enough. Someone needs to build the data pipeline to collect whatever input data the model needs -- inputs like the last 5 videos that you watched -- and that is done by a recommendation system.

Recommendation systems are not just about suggesting products to users. Sometimes, they can be about suggesting users for products.
For example, in marketing applications, you may have a new promotion and you want to find the 1000 most relevant current customers. That's called targeting, and that's also done by a recommendation system.



Many times, recommendation systems are not about what you would think of as “products”.

When Google Maps suggests a route that avoid toll roads, that’s a recommendation system.
When Smart Reply in GMail suggests possible replies to an email you received, that’s a recommendation system.

Recommendation systems are about personalization

Recommendation systems are about personalization.
It's about taking your product that works for everyone and personalizing it for an individual user.

Agenda

Introduction

Recommendation Systems

Content-based Recommendation Systems

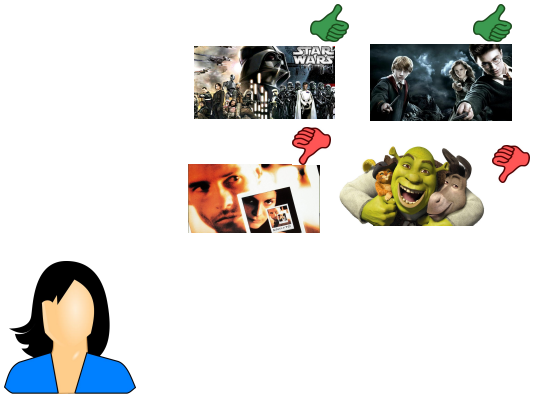
Collaborative Filtering

Neural Networks for Recommendation Systems

Building an End-to-End Recommendation System

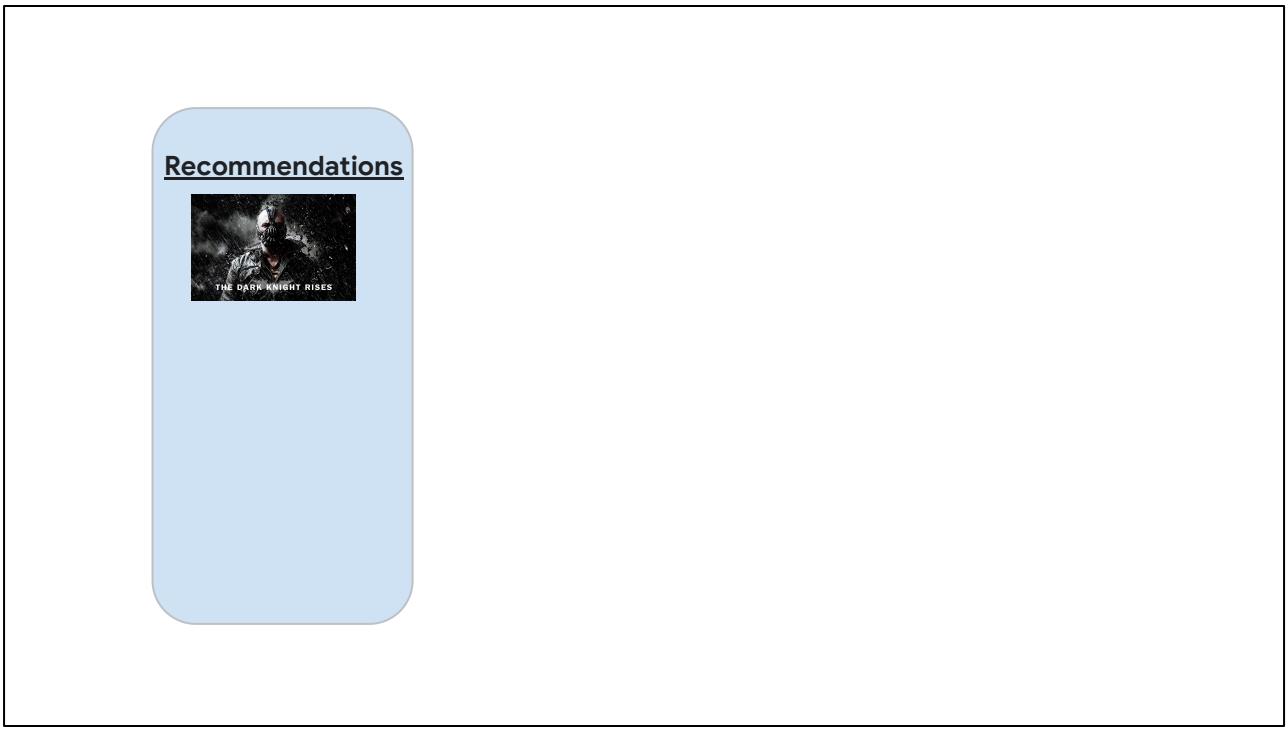
So, let's say you want to recommend movies to users.
You can do this in several ways.
The first of these is to use a content-based recommendation system.

In a content-based recommendation system, you use the metadata about your products.
For example, perhaps you know which movies are cartoons and which movies are scifi.



Now suppose you have a user who has seen and rated a few movies. Some she liked and gave a thumbs up, and some she didn't. We would like to know which movie in our database to recommend next.

Remember that we have metadata about the movies. We know that this particular user likes sci-fi and doesn’t like cartoons.
So, we might use that to recommend popular sci-fi dramas to this user.



So, perhaps we recommend The Dark Knight Rises.

Notice that this recommendation is based on *knowing* something about the content. You are simply recommending the most popular items in a category that the user likes.
Maybe you don't even have individual users's preferences. All you might have is a market segmentation -- which movies are liked by users in which regions of the country.
And that is enough to build a content-based recommendation system.

Arguably, there is no machine learning here -- it's a simple rule that relies on the builder of the recommendation system to assign proper tags to items and users.

Agenda

Introduction

Recommendation Systems

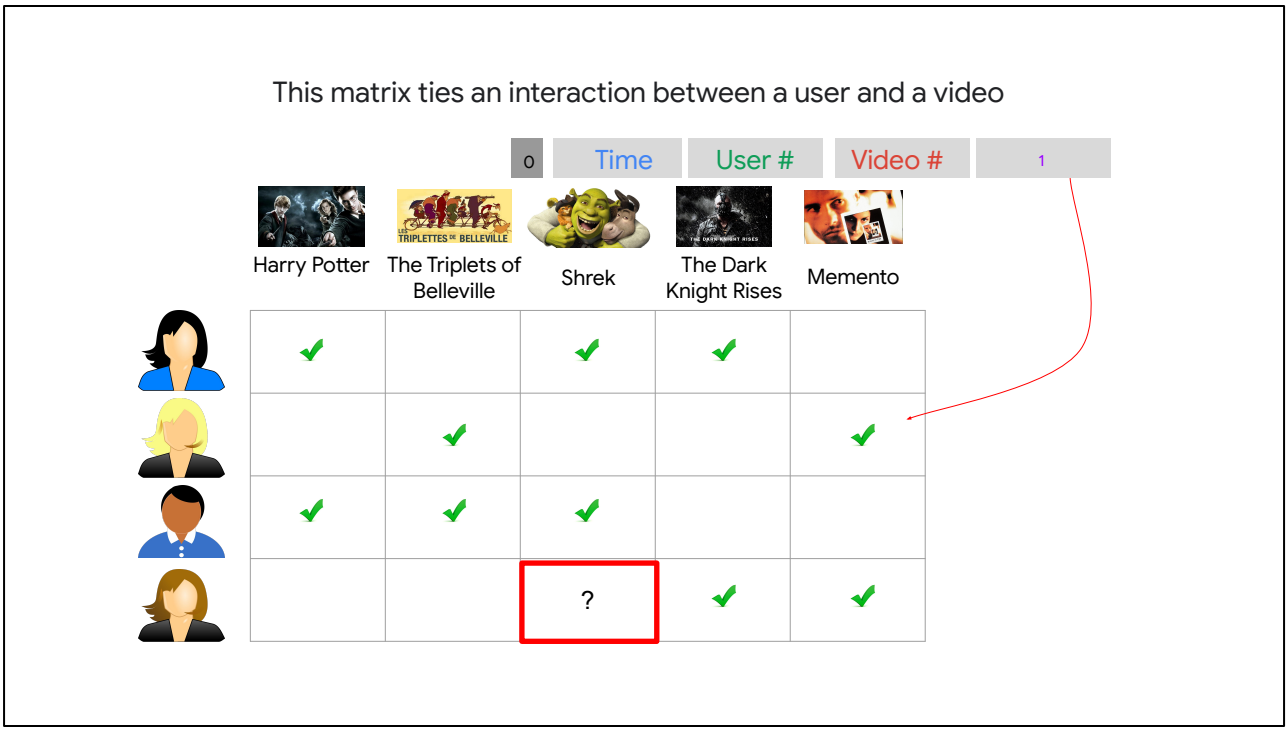
Content-based Recommendation Systems

Collaborative Filtering

Neural Networks for Recommendation Systems

Building an End-to-End Recommendation System

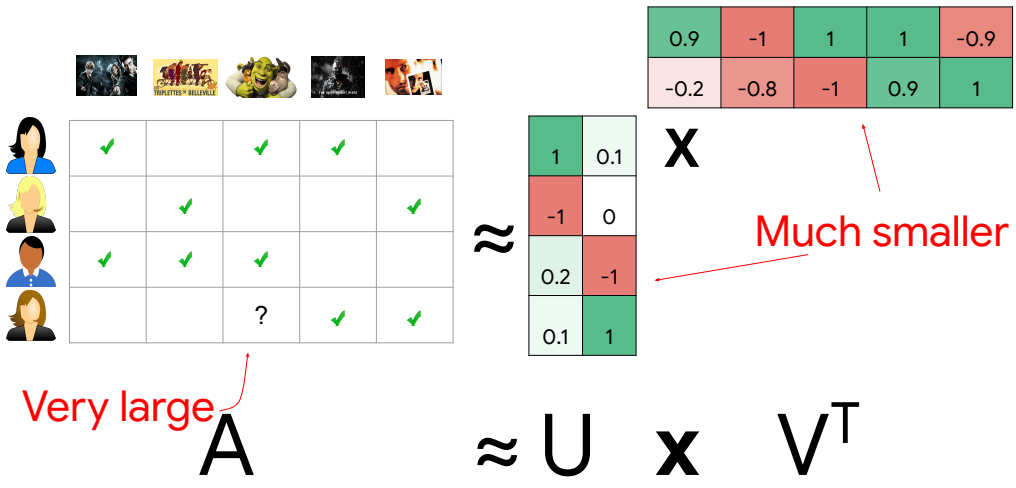
That’s a content-based recommendation system.
In collaborative filtering, you don’t have any metadata about the products.
Instead, you learn about item similarity and user similarity from the ratings data itself.



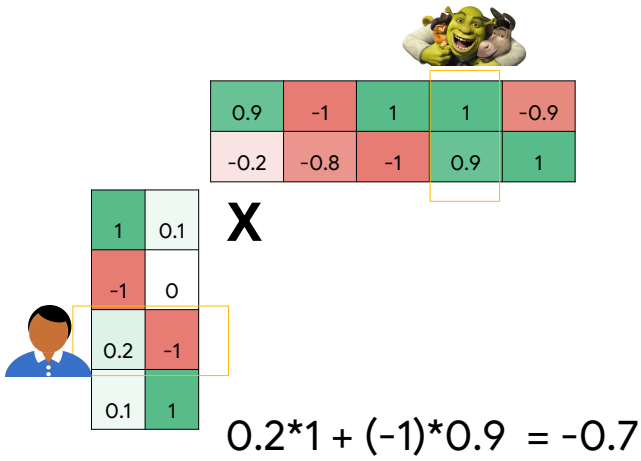
We might store our user-movie data in a matrix like this, with checkmarks indicating whether the user watched the complete movie or commented on it or gave it a star rating or however it is that we measure that a specific user liked a specific movie.

This matrix is very large -- since you might have millions to billions of users, and hundred to millions of movies.
Any individual will tend to have watched only a handful of these movies, so most of this matrix is sparse.

The factorization splits this matrix into row factors and column factors that are essentially user and item embeddings



The idea behind collaborative filtering that the very large, very sparse user-by-item matrix can be approximated by the product of two smaller matrices -- of user factors and item factors.



Then, if we need to find whether a user will like a particular movie, it is as simple as taking the row corresponding to the user and the column corresponding to the movie and multiplying them to get the predicted rating.

To recommend movies to users, we recommend the movies that we predict they will rate the highest.

The WALS Estimator in TensorFlow does not need any labels; it just needs the ratings matrix organized into [illegible]

Sparse and large

```
def training_input_fn():
  features = {
    INPUT_ROWS: tf.SparseTensor(...)
    INPUT_COLS: tf.SparseTensor(...)
  }

  return features, None
```

Shouldn't feeding the rows be enough? Why also columns?

Of course, you will learn how to implement collaborative filtering in TensorFlow.

Agenda

Introduction

Recommendation Systems

Content-based Recommendation Systems

Collaborative Filtering

Neural Networks for Recommendation Systems

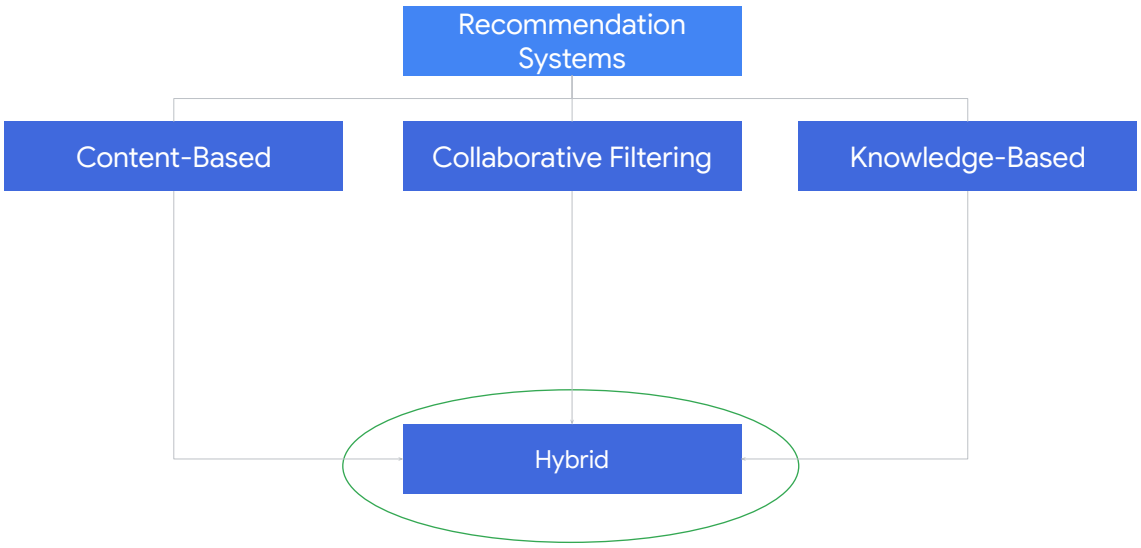
Building an End-to-End Recommendation System

One of the really cool things about collaborative filtering is that you don't need to know any metadata about your items.
You also don't need to do market segmentation of your users.
As long as you have an interactions matrix, you are ready to go.

If you do have metadata about users and items, you can use content-based recommendation systems.

What if you have both? You have metadata, and you have an interactions matrix?

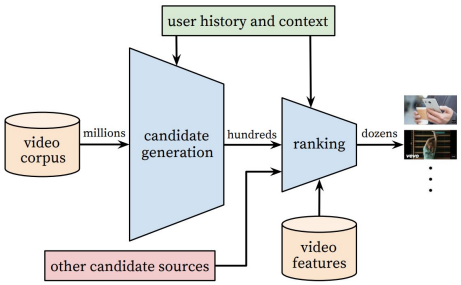
Real-world recommendation systems are a hybrid of three broad theoretical approaches



Then, you can use neural networks to combine all of the advantages, and eliminate the disadvantages of all three types of recommendation systems.
Three types? Well, there is actually a third type of recommendation system, called knowledge-based, that can be used to provide business impact inputs to systems. We'll cover this in the course.

The hybrid models use all the data available and connect all of these models together into an ML pipeline.

YouTube video recommendations



Incidentally, this is how YouTube works. We'll discuss some of the inner workings of the YouTube video hybrid recommendation system as an inspiration to building your own hybrid model.

Agenda

Introduction

Recommendation Systems

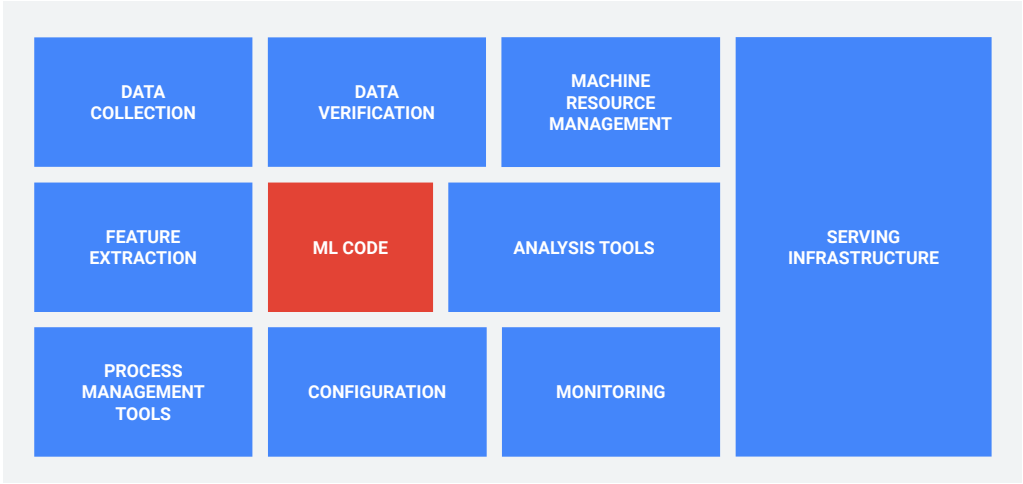
Content-based Recommendation Systems

Collaborative Filtering

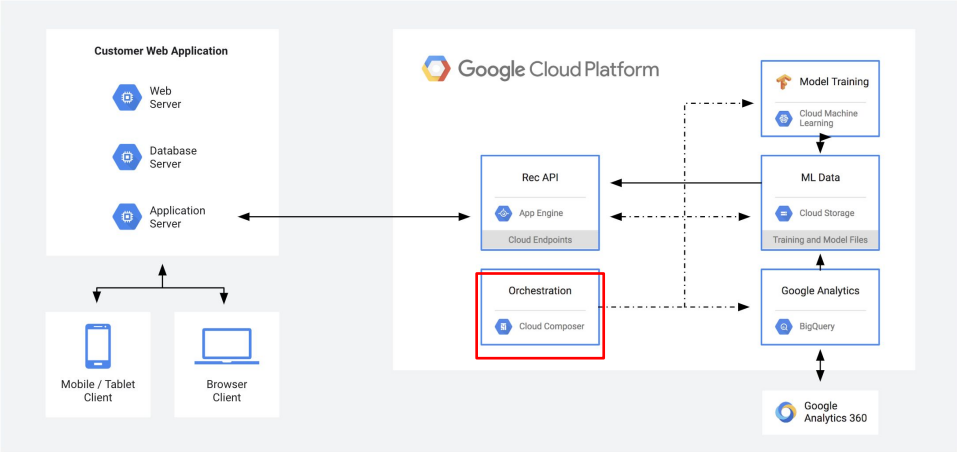
Neural Networks for Recommendation Systems

Building an End-to-End Recommendation System

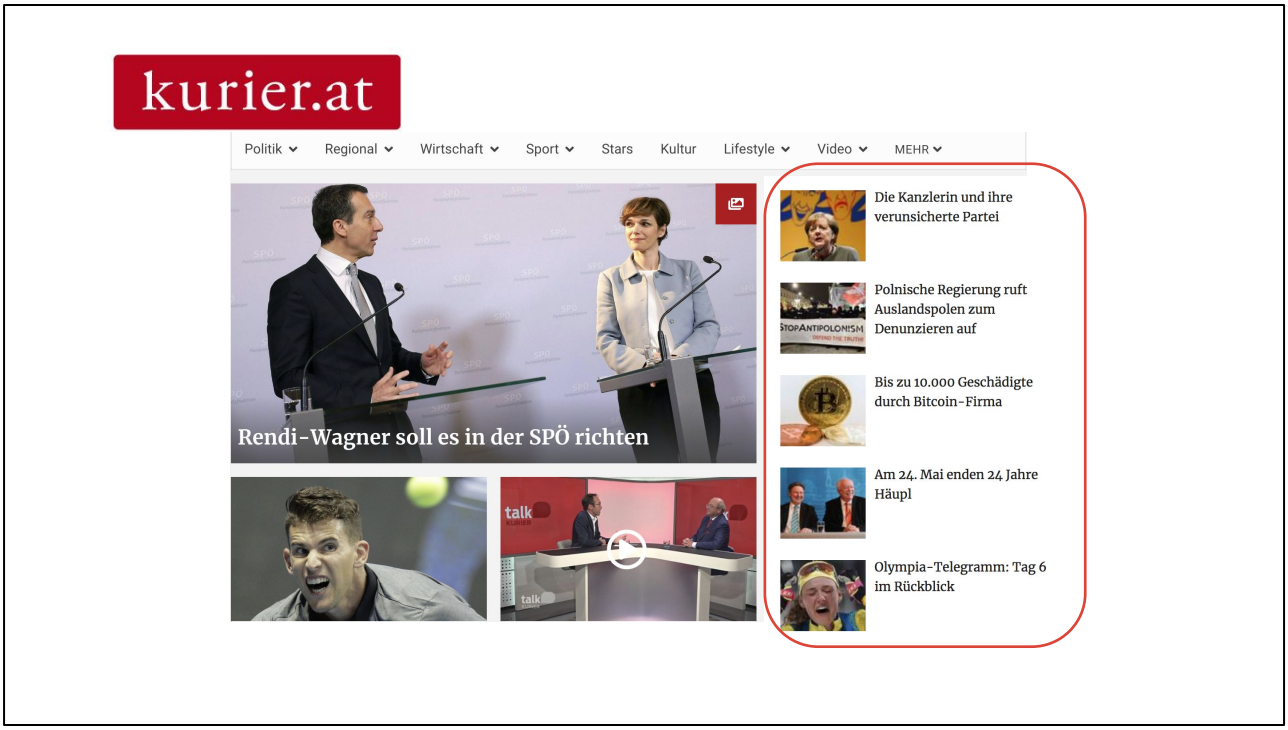
Finally, we will see how we can productionize and automate much of the necessary pipeline using the greater Google Cloud Platform ecosystem.



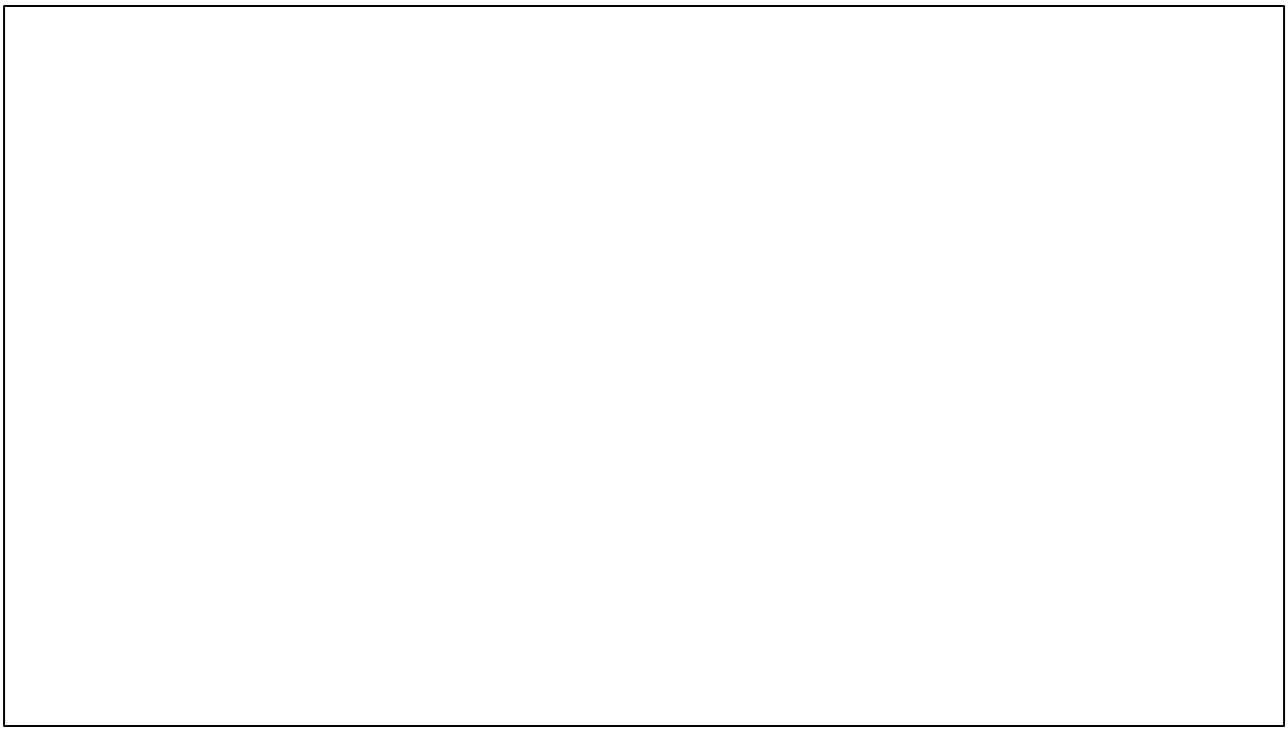
Because, as we know, from course 7 of the machine learning series, the machine learning model is only a small part of the overall system.



Consequently, we will look at the architecture of an end-to-end system for recommendations, so that we can orchestrate the continuous retraining of the recommendation system as new ratings data comes in from users.



The real-world example that we'll use is to use a week of data extracted from the web logs of Kurier.at, a large news provider in Austria.



Recommendation systems are often the machine learning systems that you encounter the most often in enterprise settings.
From a business impact standpoint, recommendation systems can allow you to sell products you could never sell before to users you could never reach before.

To learn how to build effective recommendation systems, though, you will have to know all the topics that we have covered in all the courses so far.
The data you have may be structured data, might be images, might be text, and you will need to bring in all the data into a powerful machine learning model.

We have quite a journey ahead of us. Let’s get started!