Data Pipelines

1. Data Pipelines

00:00 - 00:14

Nice job so far! Let's learn about data pipelines. So far we've learned about data collection and storage, but how can we scale all this? This is where data pipelines come in.

2. Data collection and storage

00:14 - 00:27

Data engineers work to collect and store data, so that others, like analysts and data scientists can access data for their work, whether it's for visualization or building machine learning models.

3. How do we scale?

00:27 - 01:00

But how do we scale this? Consider the different data sources you learned about - what if we're collecting data from more than one data source? And then, what if these data sources have different types of data? For example, consider real-time streaming data, which is data that is continuously being generated, like tweets from all around the world. This makes storing this incoming data complicated, because as a data engineer, you want to make sure data is organized and easy to access.

4. What is a data pipeline?

01:00 - 02:23

Enter the data pipeline. A data pipeline moves data into defined stages, for example, from data ingestion through an API to loading data into a database. A key feature is that pipelines automate this movement. Data is constantly coming in and it would be tedious to ask a data engineer to manually run programs to collect and store data. Instead a data engineer schedules tasks whether it's hourly, daily, or tasks can be triggered by an event. Because of this automation, data pipelines need to be monitored. Luckily, alerts can be generated automatically, for example, when 95% of storage capacity has been reach or if an API is responding with an error. Data pipelines aren't necessary for all data science projects, but they are when working with a lot of data from different sources. There isn't a set way to make a pipeline - pipelines are highly customized depending on your data, storage options, and ultimate usage of the data. ETL, which stands for extract, transform, and load, is a popular framework for data pipelines. Let's explore it with a case study.

5. Case study: smart home

02:23 - 03:03

After learning about IoT devices and APIs, you decide to try out both. Specifically, you want to use APIs and devices in your house to better understand the status of your house and neighborhood. You gather a list of data sources and associated information. The first two are provided by APIs. Every 30 minutes, you get the weather conditions, and you get Tweets geotagged in your neighborhood whenever they are published. The remaining rows are IoT devices that send their sensor data over the internet at the specified frequencies.

6. Extract

03:03 - 03:23

How does it all come together? First, we begin with extracting all the data from the data sources we listed, whether it's an API or setting up an IoT device. However, a quick look at the frequencies and structures makes us realize that storing the raw data as is won't work.

7. Transform

03:23 - 03:27

This is where the transform phase comes in.

8. Transform

03:27 - 04:24

In this stage, we transform data to make sure data stays organized so that others can easily find relevant data and use it. A common transformation is joining data from different sources into one dataset. Another is converting the incoming data's structure to fit a database's schema. A database's schema informs us how data must be structured before loading it in. A transformation can also be removing irrelevant data. For example, the Twitter API, not only gives you a tweet but others details, like the number of followers the author has, which is not useful in our scenario, so we shouldn't store it. Data gets altered throughout the data science workflow, it's important to note that analytical tasks like data preparation and exploration don't occur at this stage - we'll see this in chapter 3.

9. Load

04:24 - 04:31

Finally, we load the data into storage so that it can be used for visualization and analysis.

10. Automation

04:31 - 04:50

Once we've set up all those steps, we automate. For example, we can say every time we get a tweet, we transform it in a certain way and store it in a specific table in our database. There are tools that specialized to do this, the most popular is called Airflow.

11. Let's practice!

04:50 - 04:56

Now, it's time for some exercises.