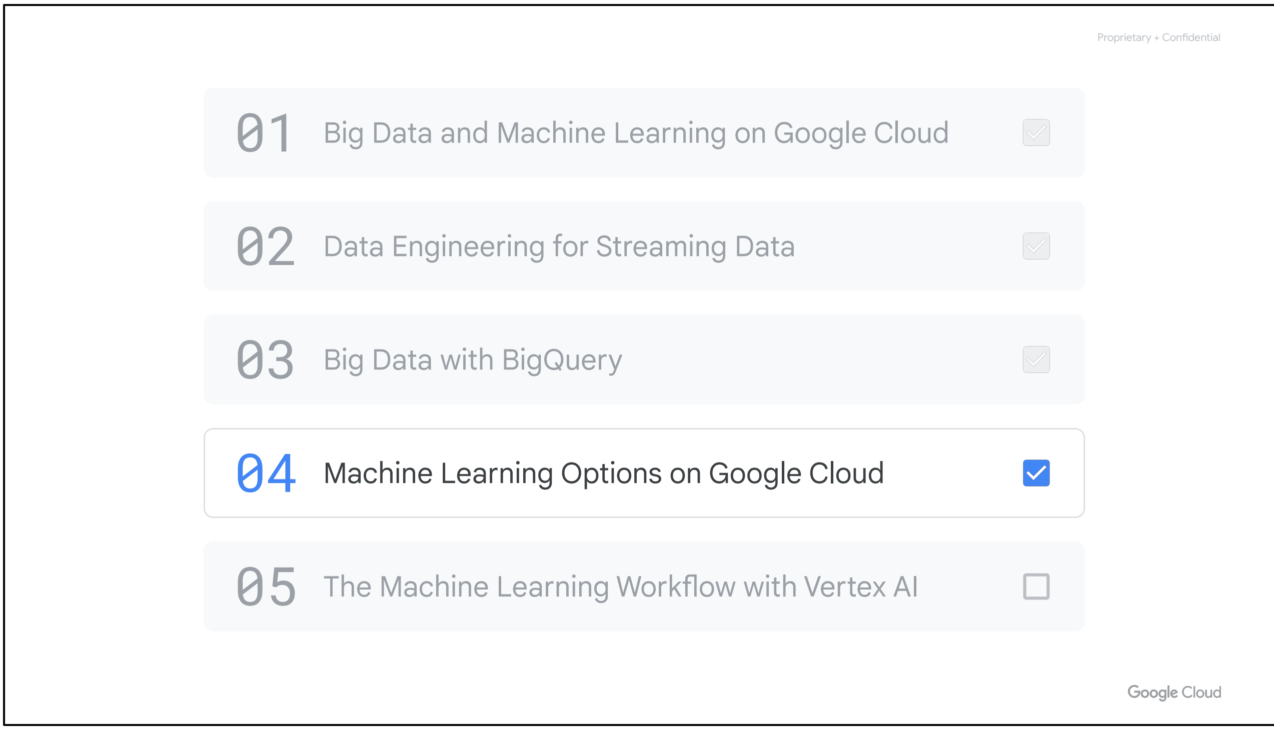
Machine Learning Options on Google Cloud

Module 4

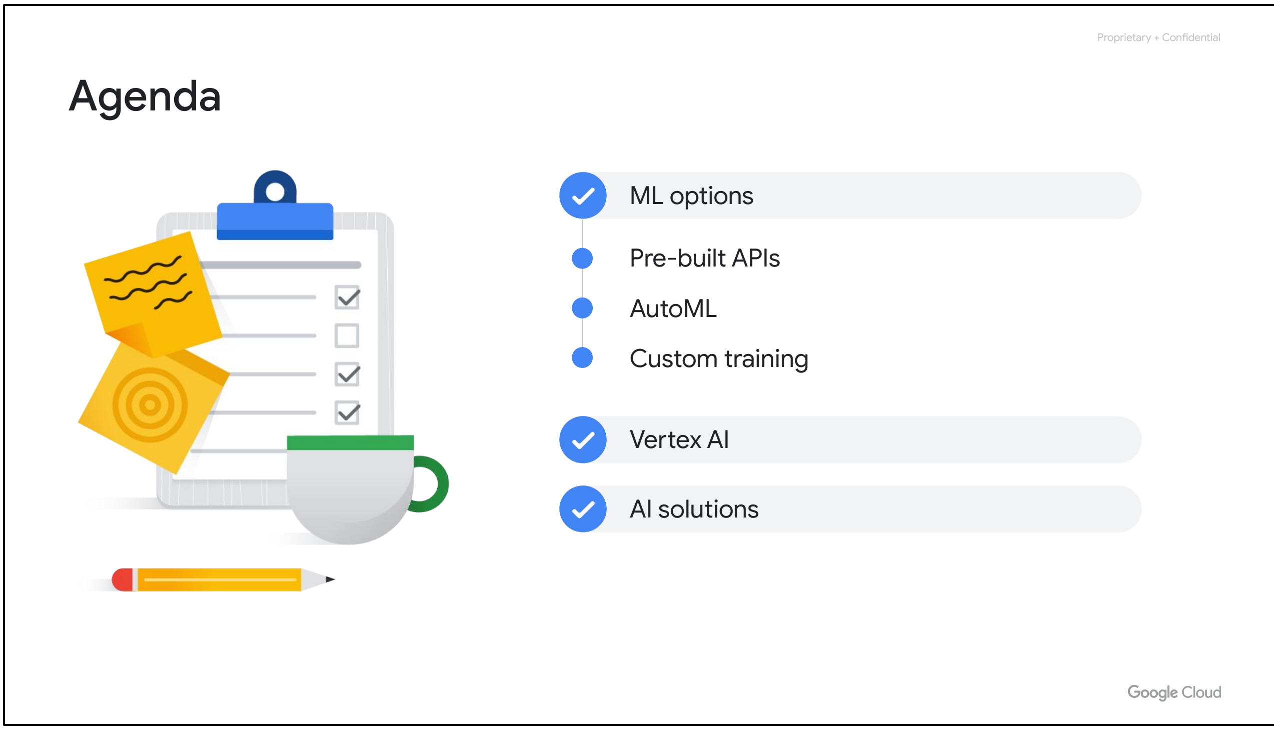
Google Cloud Big Data and Machine Learning Fundamentals

01

Introduction



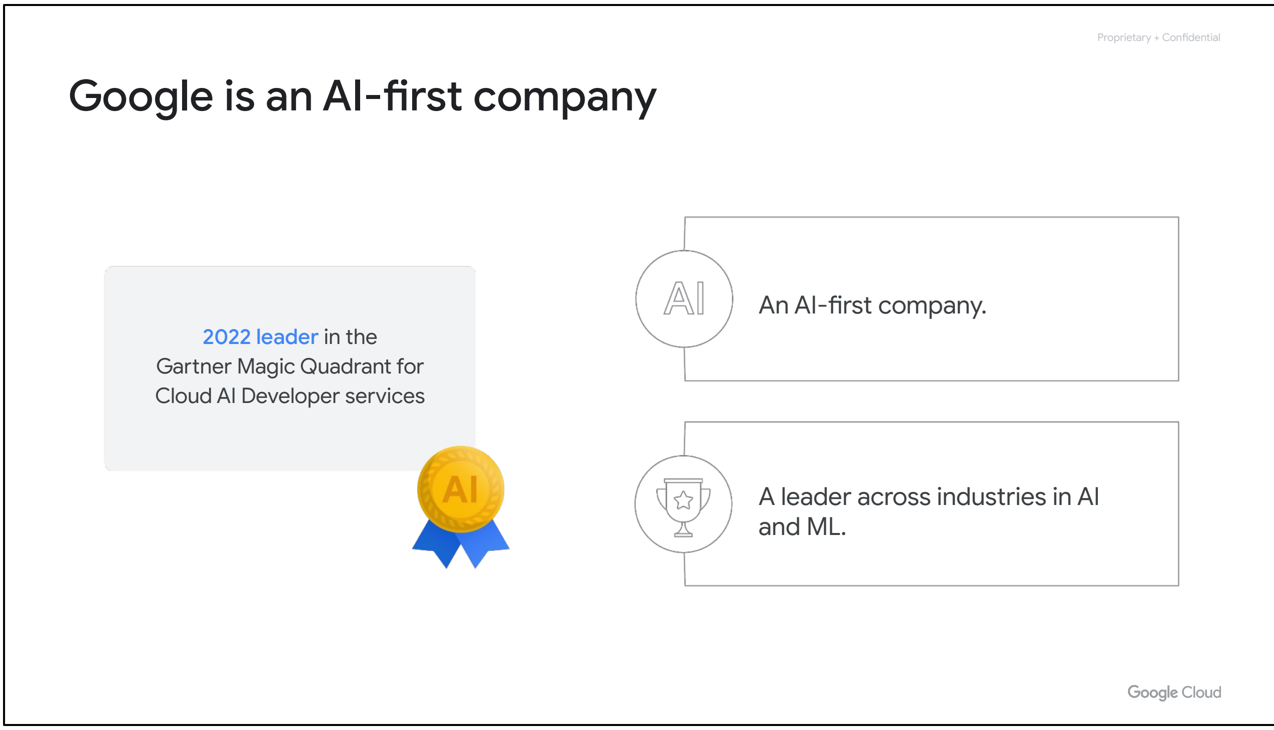
In the previous two modules of this course, you learned about many data engineering tools available from Google Cloud such as Dataflow, Pub/Sub, and Looker vs. Data Studio in module 2 and BigQuery in module 3. Now let’s switch our focus to machine learning. In the next two modules, you’ll learn ML options provided by Google in module 4 and the ML workflow in module 5.



In this module, we'll explore the different options Google Cloud offers for building machine learning models, specifically, we'll introduce pre-built APIs, AutoML, and custom training.

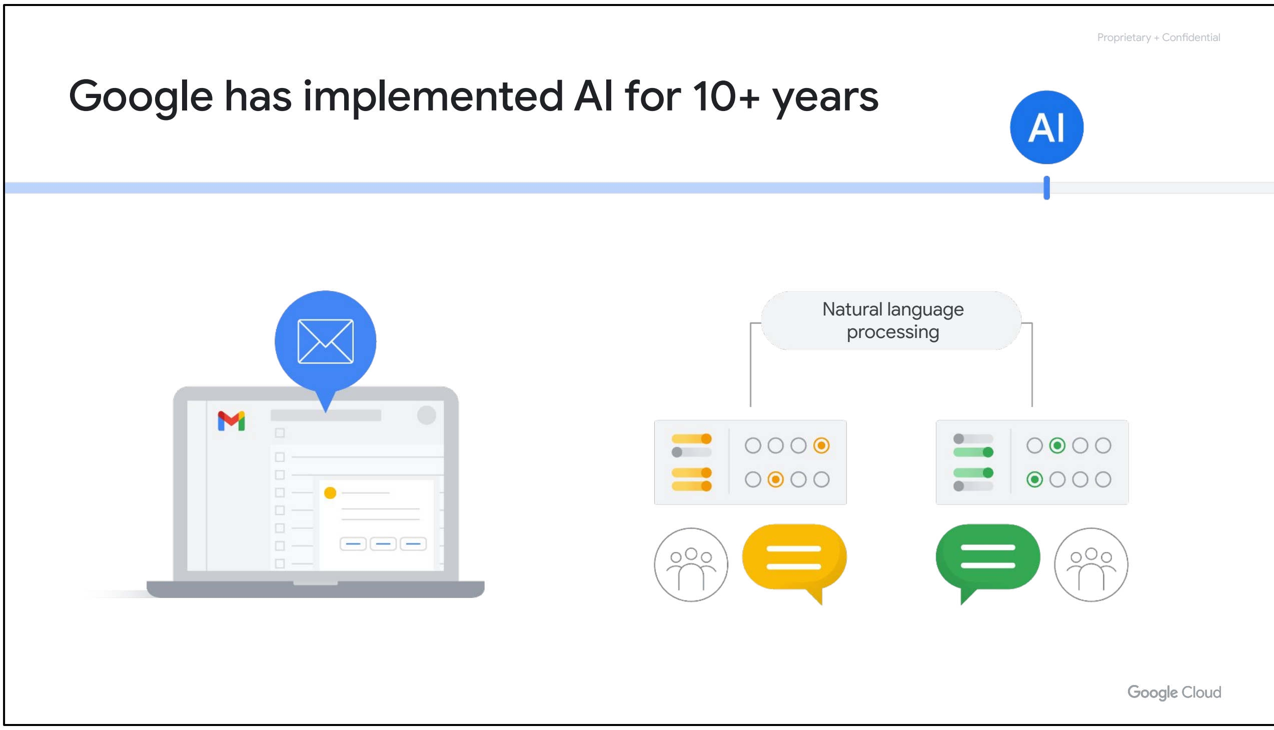
Additionally, we will explain how a product called Vertex AI can help solve machine learning challenges.

In the end, we will introduce the Google Cloud AI solutions from both horizontal (meaning across industries) and vertical (meaning industry specific) perspectives.



So you might be wondering, “Why should I trust Google for artificial intelligence and machine learning?”

Google is an AI-first company, and is recognized as a leader across industries because of its contributions in the fields of artificial intelligence and machine learning. In 2022 Google was recognized as a leader in the Gartner Magic Quadrant for Cloud AI Developer services, and in recent years has also received recognition in numerous annual industry awards and reports.



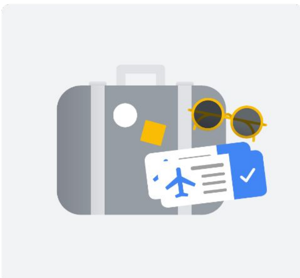
And at Google we've been implementing artificial intelligence for over ten years into many of our critical products, systems, and services.

For example, have you ever noticed how Gmail automatically suggests three responses to a received message? This feature is called Smart Reply, which uses artificial intelligence to predict how you might respond. Behind this intelligence is AI technology known as natural language processing, which is just one example of an impressive list of technologies that Google scientists and engineers are working on. We'll explore these in more depth later in the course.

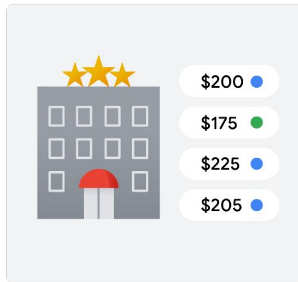
Google's goal is to enable every company to be an AI company

The goal of these technologies is not for exclusive use to only benefit Google customers. The goal is to enable every company to be an AI company by reducing the challenges of AI model creation to only the steps that require human judgment or creativity.

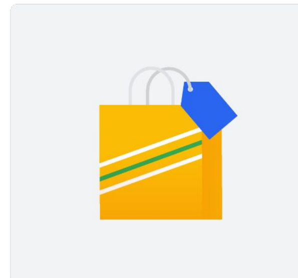
AI in different industries



Scheduling



Pricing



Inventory planning

Google Cloud

So for workers in the travel and hospitality field, this might mean using AI and ML to improve aircraft scheduling or provide customers with dynamic pricing options. For retail-sector employees, it might mean using AI and ML to leverage predictive inventory planning. The potential solutions are endless.

Proprietary + Confidential

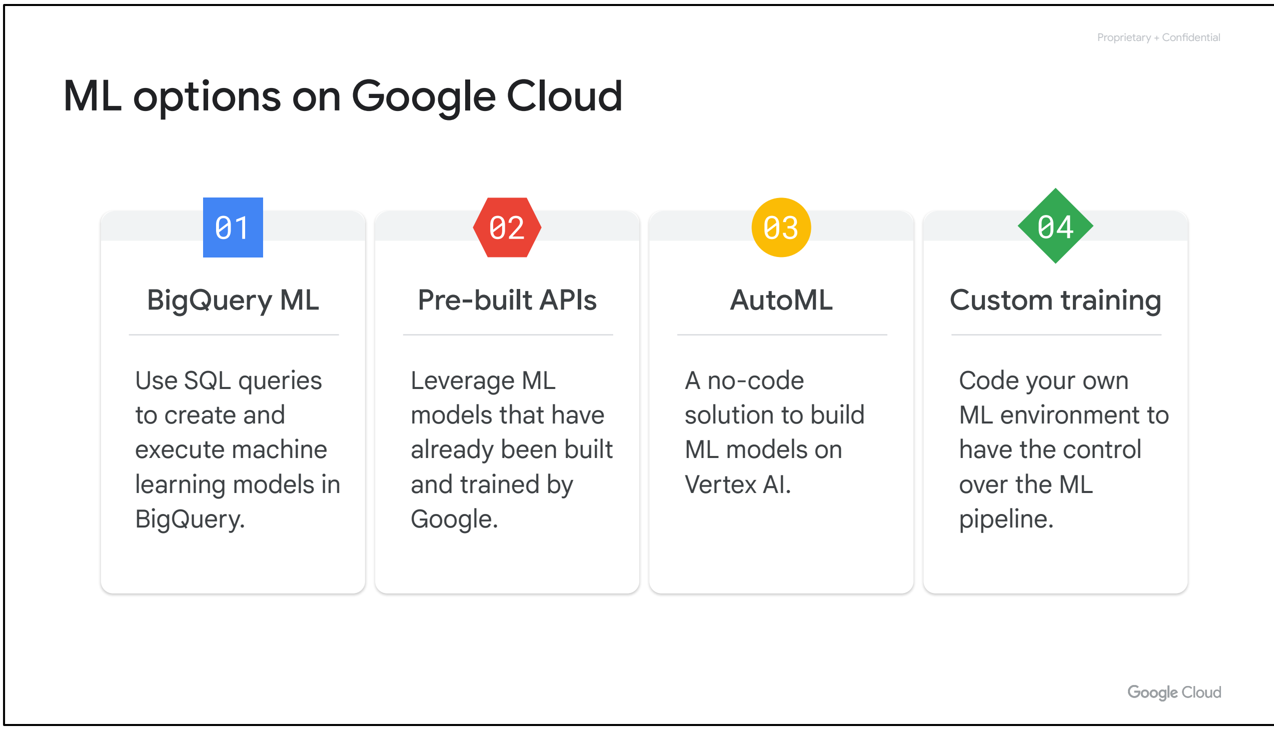
What are the problems in your business that artificial intelligence and machine learning might **help you solve**?

Google Cloud

What are the problems in your business that artificial intelligence and machine learning might help you solve? Take a moment to think about this question before continuing to the next section.

02

Options to build ML models



Google Cloud offers four options for building machine learning models.

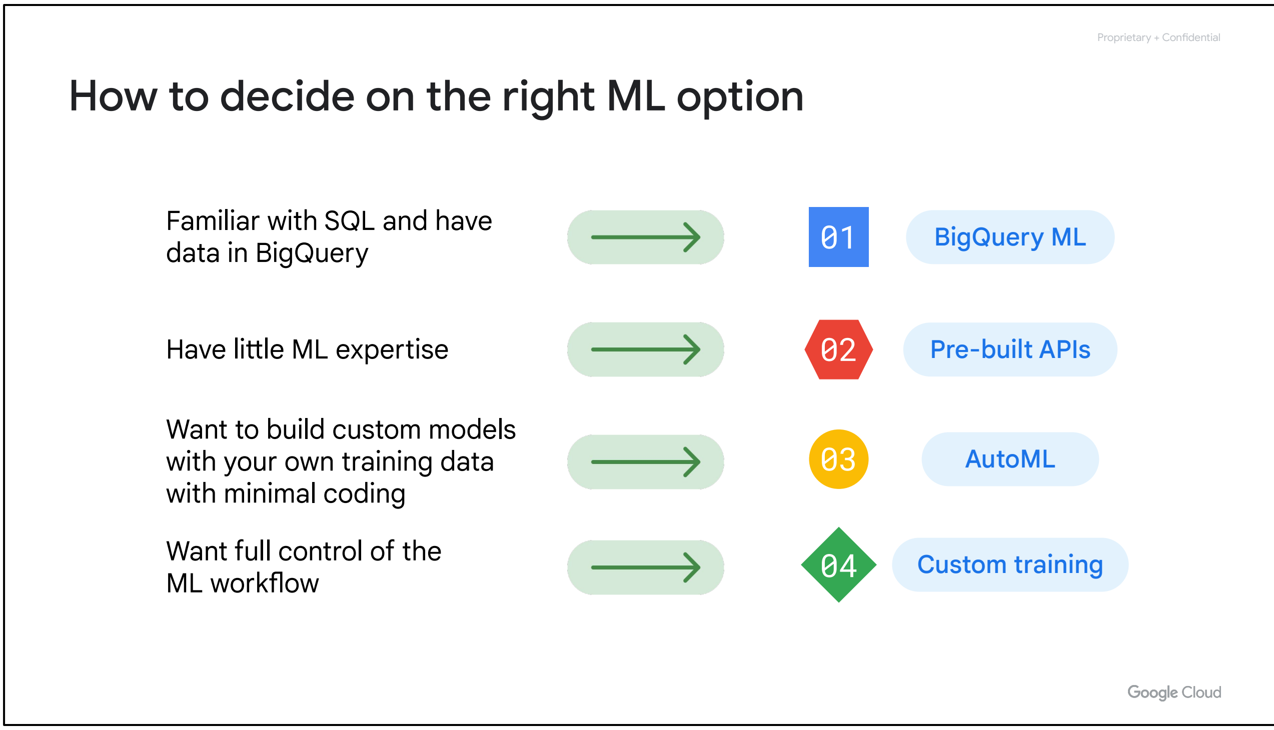
- The first option is BigQuery ML. You'll remember from an earlier module of this course that BigQuery ML is a tool for using SQL queries to create and execute machine learning models in BigQuery. If you already have your data in BigQuery and your problems fit the pre-defined ML models, this could be your choice.
- The second option is to use **pre-built APIs**, which are application programming interfaces. This option lets you leverage machine learning models that have already been built and trained by Google, so you don't have to build your own machine learning models if you don't have enough training data or sufficient machine learning expertise in-house.
- The third option is AutoML, which is a no-code solution, so you can build your own machine learning models on Vertex AI through a point-and-click interface.
- And finally, there is custom training, through which you can code your very own machine learning environment, the training, and the deployment, which gives you flexibility and provides the control over the ML pipeline..

Comparing the four ML options

	01 BigQuery ML	02 Pre-built APIs	03 AutoML	04 Custom training
Data type	Tabular	Tabular, image, text, and video	Tabular, image, text, and video	Tabular, image, text, and video
Training data size	Medium to large	No data required	Small to medium	Medium to large
ML and coding expertise	Medium	Low	Low	High
Flexibility to tune hyperparameters	Medium	None	None	High
Time to train a model	Medium	None	Medium	Long

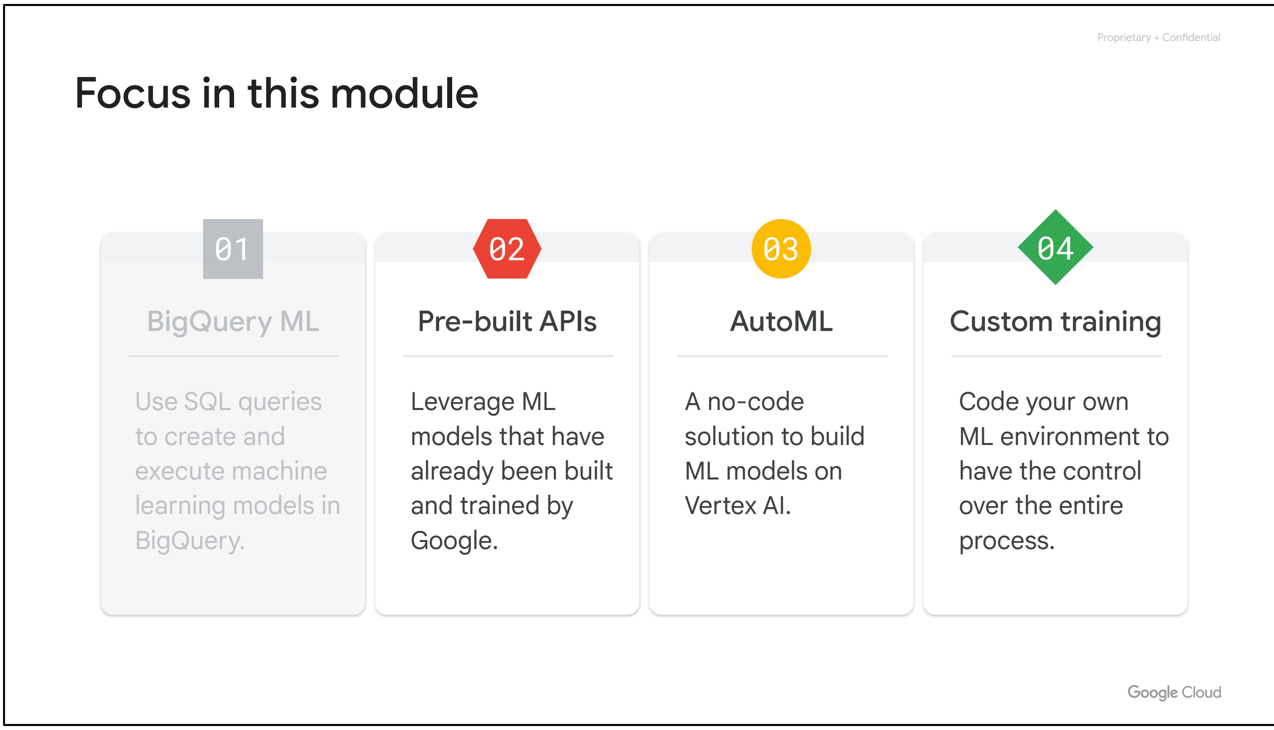
Let's compare the four options to help you decide which one to use for building your ML model. Please note that the technologies change constantly and this is only a brief guideline.

- Data type: BigQuery ML only supports tabular data while the other three support tabular, image, text, and video.
- Training data size: Pre-built APIs do not require any training data, while BigQuery ML and custom training require a large amount of data.
- Machine learning and coding expertise: Pre-Built APIs and AutoML are user friendly with low requirements, while Custom training has the highest requirement and BigQuery ML requires you to understand SQL.
- Flexibility to tune the hyperparameters: At the moment, you can't tune the hyperparameters with Pre-built APIs or AutoML, however, you can experiment with hyperparameters using BigQuery ML and custom training.
- Time to train the model: Pre-built APIs require no time to train a model because they directly use pre-built models from Google. The time to train a model for the other three options depends on the specific project. Normally, custom training takes the longest time because it builds the ML model from scratch, unlike AutoML and BigQuery ML. AutoML uses a backend technology called *Transfer Learning* (you'll learn it in the later section), meaning to train a new ML model based on existing training results to speed the model training time. Custom training, compared to AutoML, has to train a model from scratch, which normally takes longer time.



Selecting the best option will depend on your business needs and ML expertise.

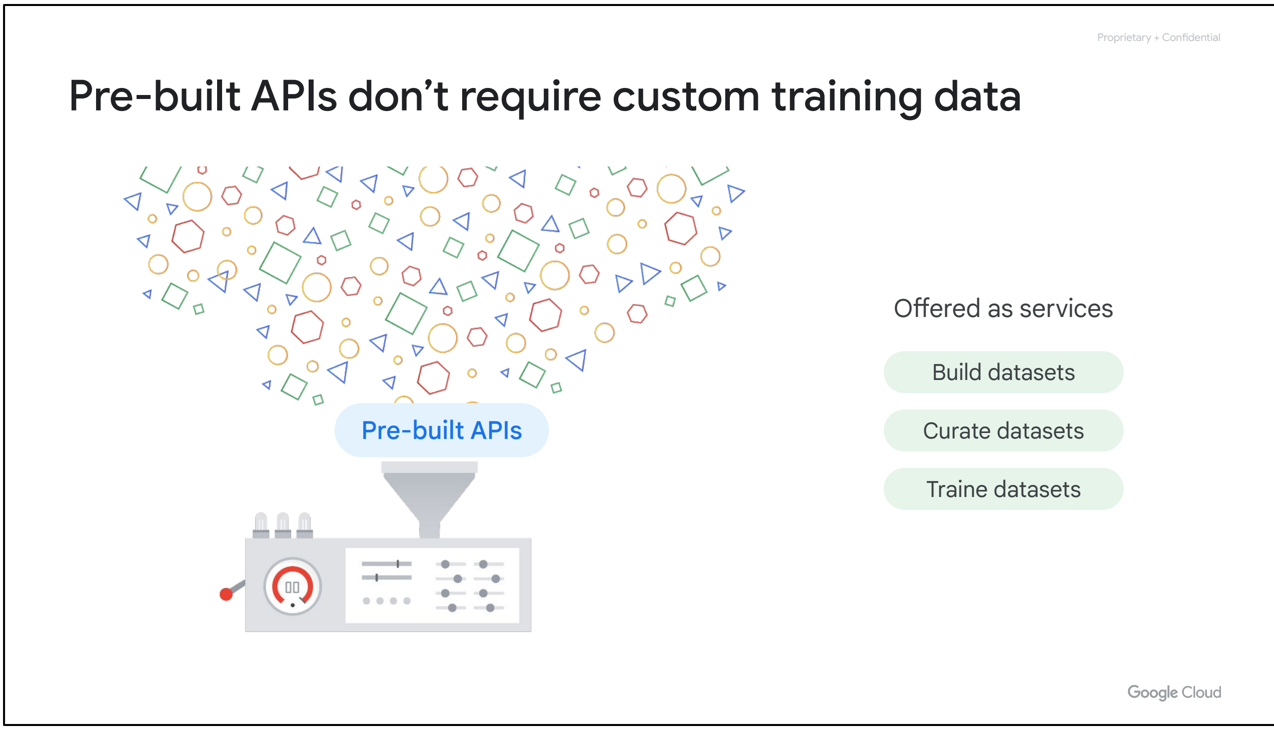
- If your data engineers, data scientists, and data analysts are familiar with SQL and already have your data in BigQuery, BigQuery ML lets you develop SQL-based models.
- If your business users or developers have little ML experience, using pre-built APIs is likely the best choice. Pre-built APIs address common perceptual tasks such as vision, video, and natural language. They are ready to use without any ML expertise or model development effort.
- If your developers and data scientists want to build custom models with your own training data while spending minimal time coding, then AutoML is your choice. AutoML provides a code-less solution to enable you to focus on business problems instead of the underlying model architecture and ML provisioning.
- If your ML engineers and data scientists want full control of ML workflow, Vertex AI custom training lets you train and serve custom models with code on Vertex Workbench.



We've already explored BigQuery ML, so in the videos that follow, we'll explore the other three options in more detail.

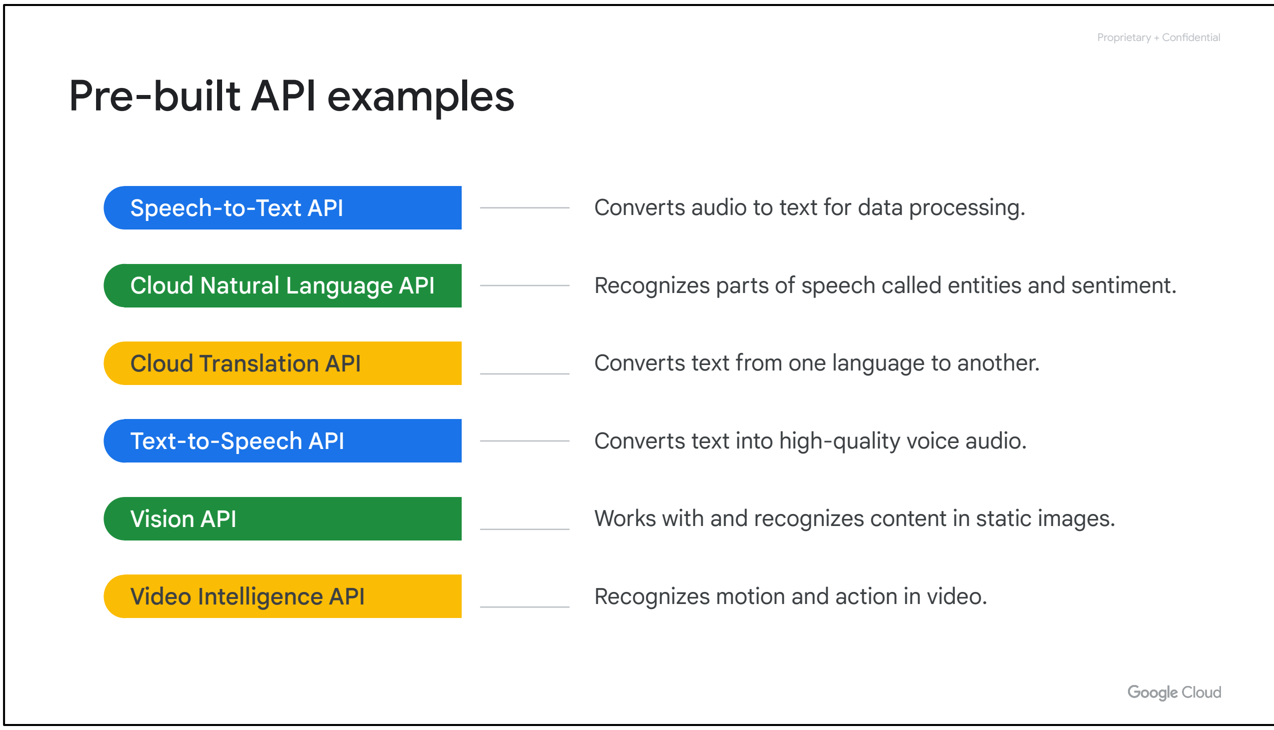
03

Pre-built APIs



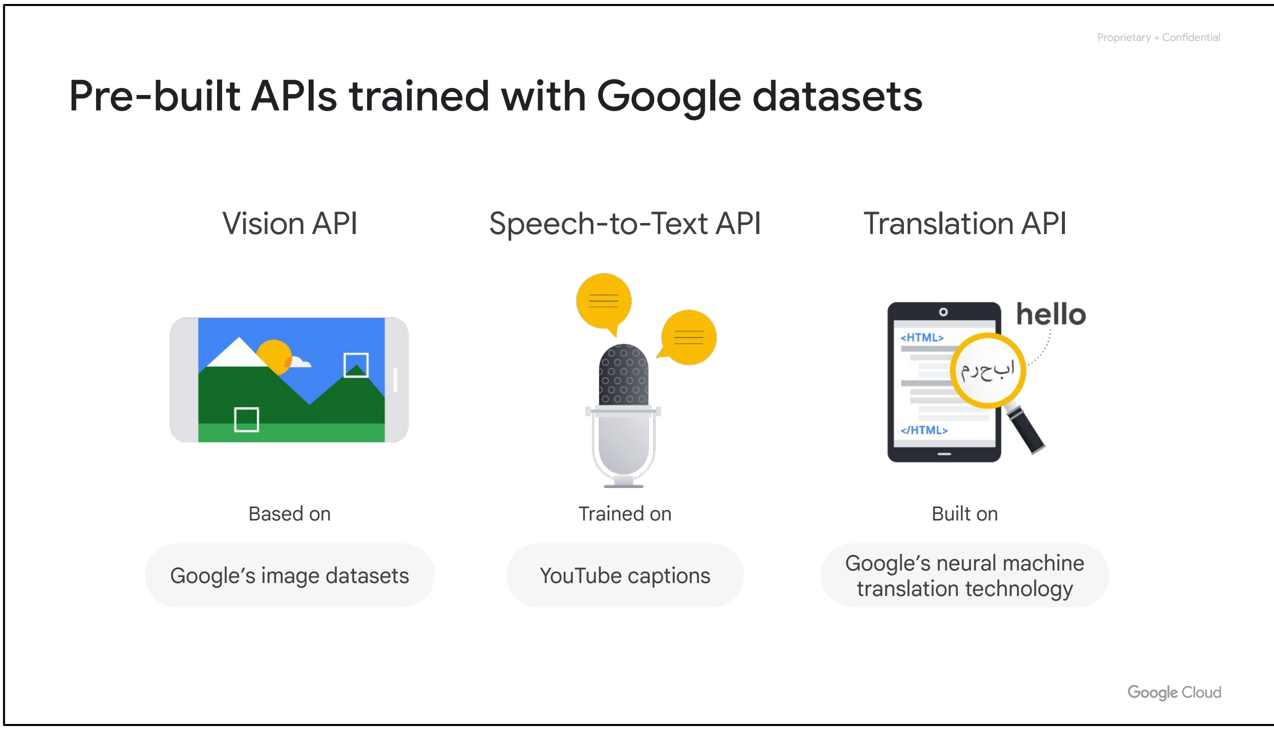
Good Machine Learning models require lots of high-quality training data. You should aim for hundreds of thousands of records to train a custom model. If you don't have that kind of data, pre-built APIs are a great place to start.

Pre-built APIs are offered as services. In many cases they can act as building blocks to create the application you want without expense or complexity of creating your own models. They save the time and effort of building, curating, and training a new dataset so you can just jump right ahead to predictions.



So, what are some of the pre-built APIs? Let’s explore a short list.

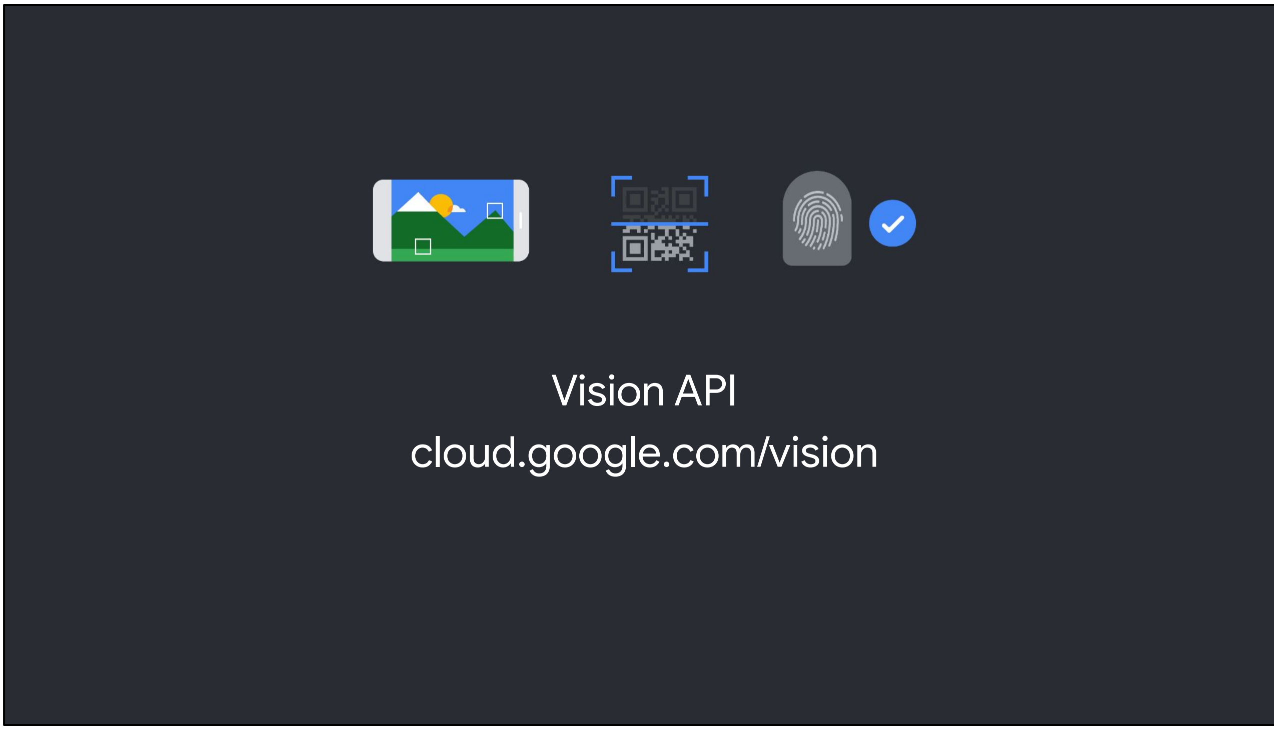
- The **Speech-to-Text API** converts audio to text for data processing.
- The **Cloud Natural Language API** recognizes parts of speech called entities and sentiment.
- The **Cloud Translation API** converts text from one language to another.
- The **Text-to-Speech API** converts text into high quality voice audio.
- The **Vision API** works with and recognizes content in static images.
- And the **Video Intelligence API** recognizes motion and action in video.



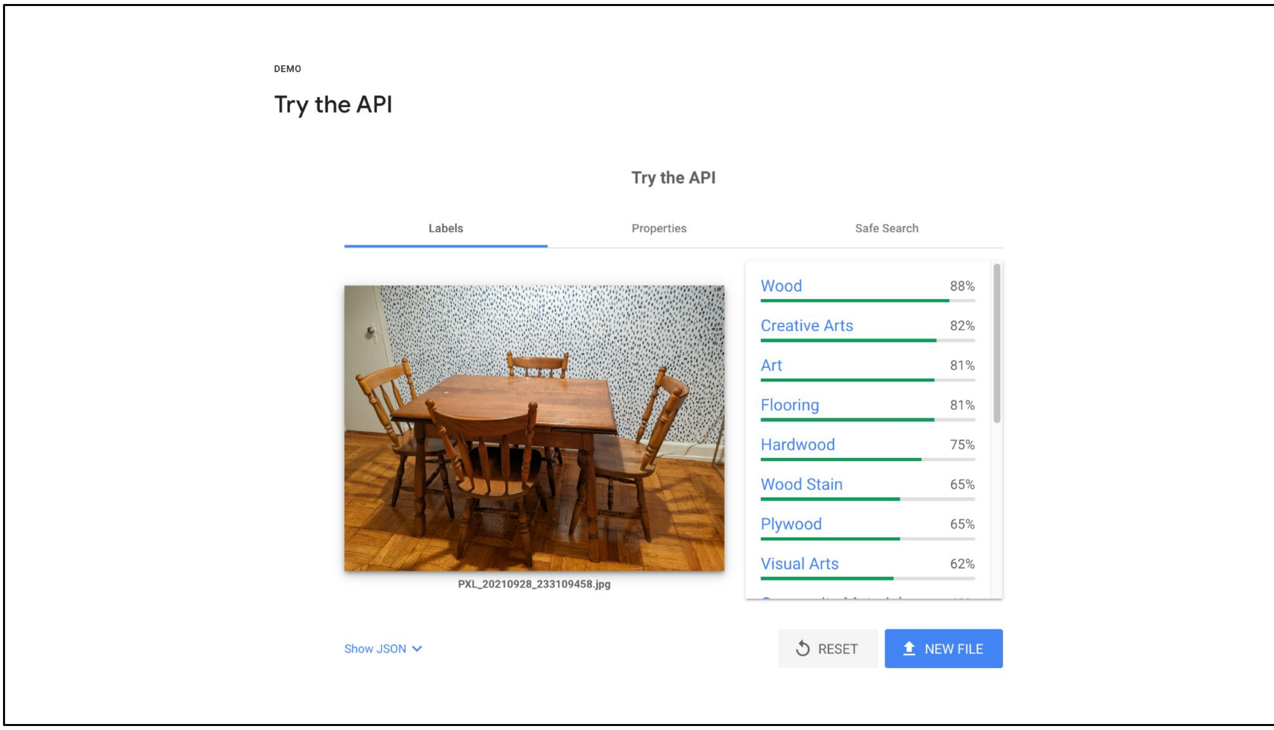
And Google has already done a lot of work to train these models using Google datasets. For example,

- the Vision API is based on Google's image datasets,
- the Speech-to-Text API is trained on YouTube captions,
- and the Translation API is built on Google's neural machine translation technology.

You'll recall that how well a model is trained depends on how much data is available to train it. As you might expect, Google has a lot of images, text, and ML researchers to train its pre-built models. This means less work for you.



Let's take a minute and try out the Vision API in a browser. Start by navigating to cloud.google.com/vision in Chrome,



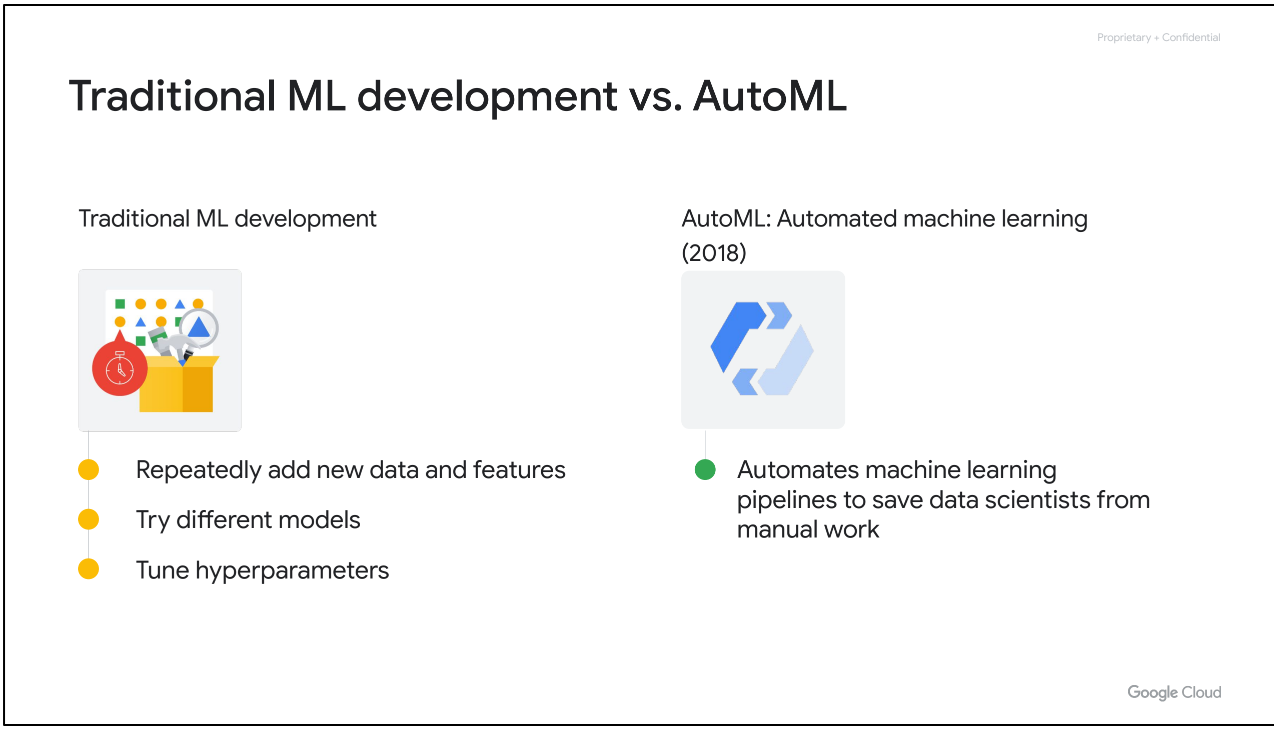
And then scroll down to try the API by uploading an image.

You can actually experiment with each of the ML APIs in a browser.

When you're ready to build a production model, you'll need to pass a JSON object request to the API and parse what it returns.

04

AutoML

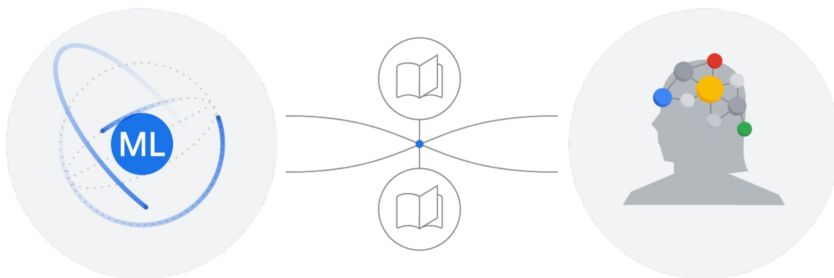


To understand AutoML, which is short for automated machine learning, let's briefly look at how it was built.

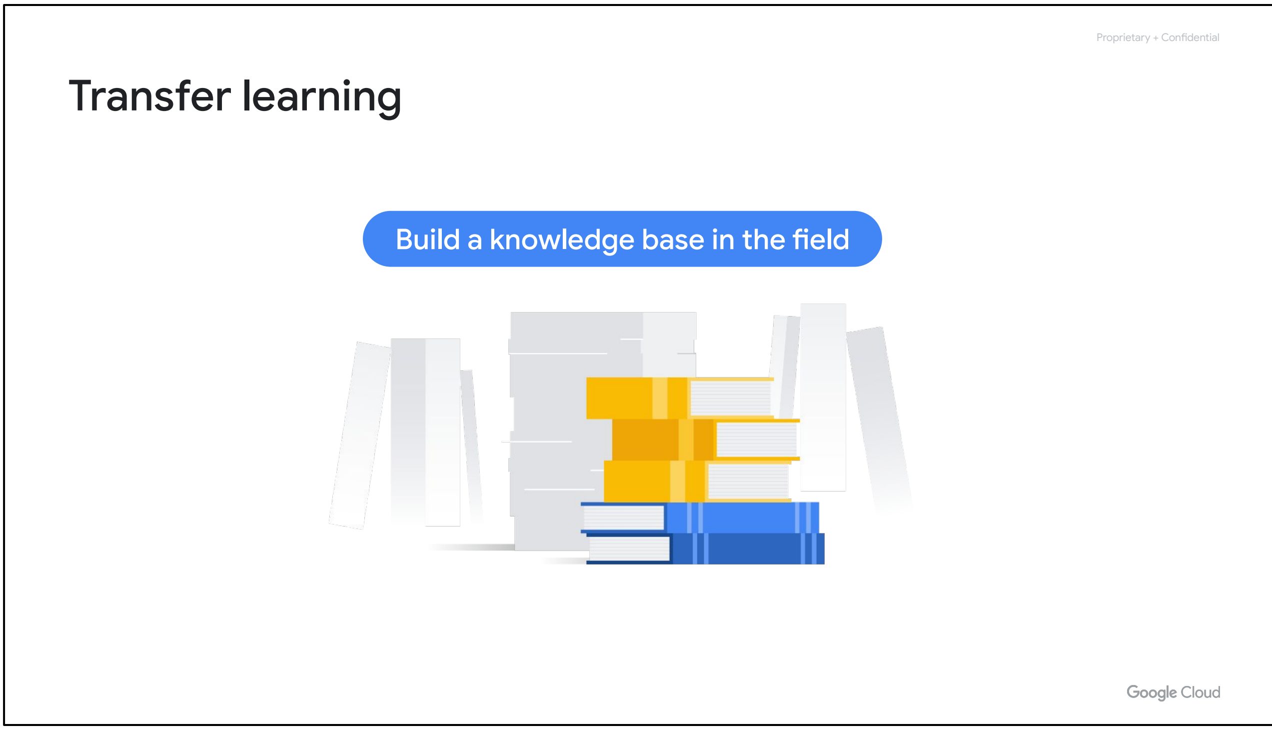
If you've worked with ML models before, you know that training and deploying ML models can be extremely time consuming, because you need to repeatedly add new data and features, try different models, and tune parameters to achieve the best result.

To solve this problem, when AutoML was first announced in January of 2018, the goal was to automate machine learning pipelines to save data scientists from manual work, such as tuning hyperparameters and comparing against multiple models.

Machine learning is similar to human learning

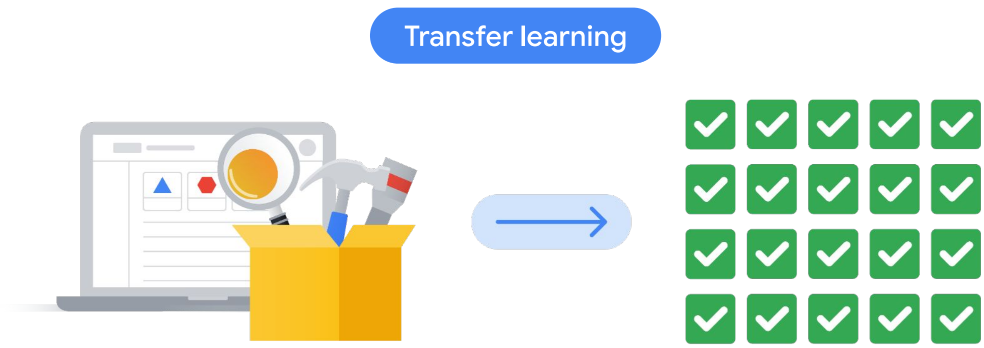


But how could this be done? Well, machine learning is similar to human learning. It all starts with gathering the right information.



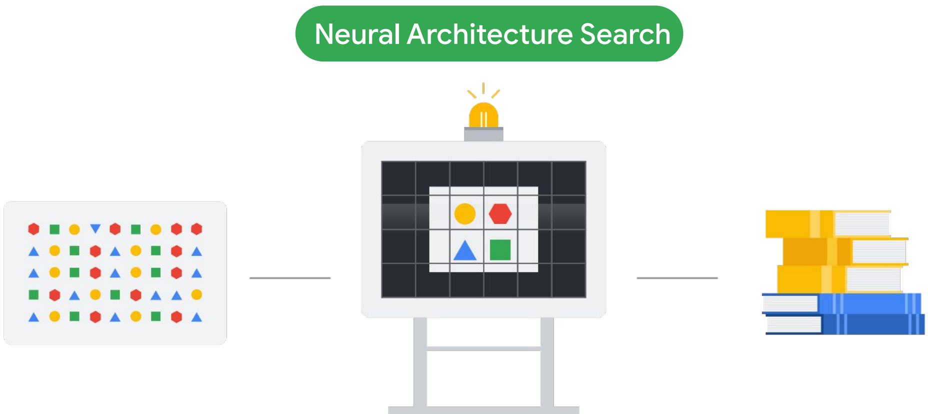
For AutoML, two technologies are vital. The first is known as **transfer learning**. With transfer learning, you build a knowledge base in the field. You can think of this like gathering lots of books to create a library.

Transfer learning from pre-trained models



Transfer learning is a powerful technique that lets people with smaller datasets, or less computational power, achieve state-of-the-art results by taking advantage of pre-trained models that have been trained on similar, larger data sets. Because the model learns via transfer learning, it doesn't have to learn from scratch, so it can generally reach higher accuracy with much less data and computation time than models that don't use transfer learning.

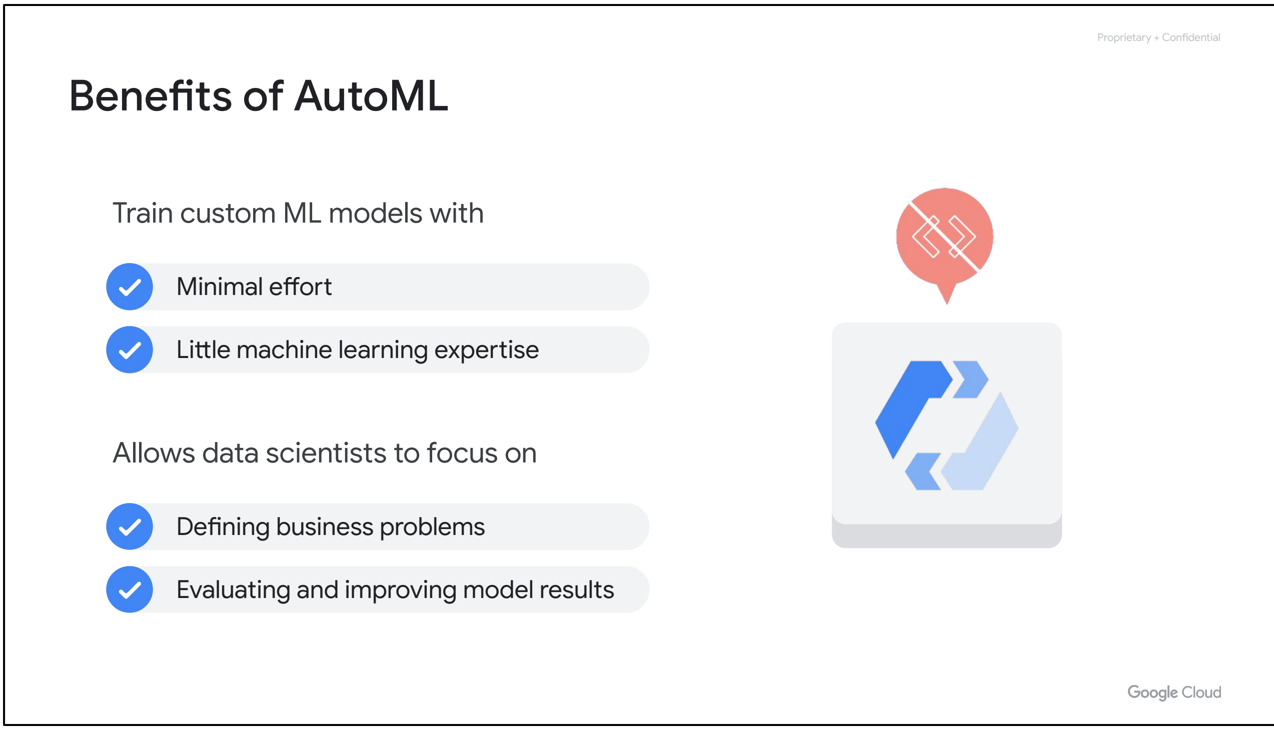
Neural Architecture Search finds the optimal model



The second technology is **neural architecture search**.The goal of neural architecture search is to find the optimal model for the relevant project. Think of this like finding the best book in the library to help you learn what you need to.

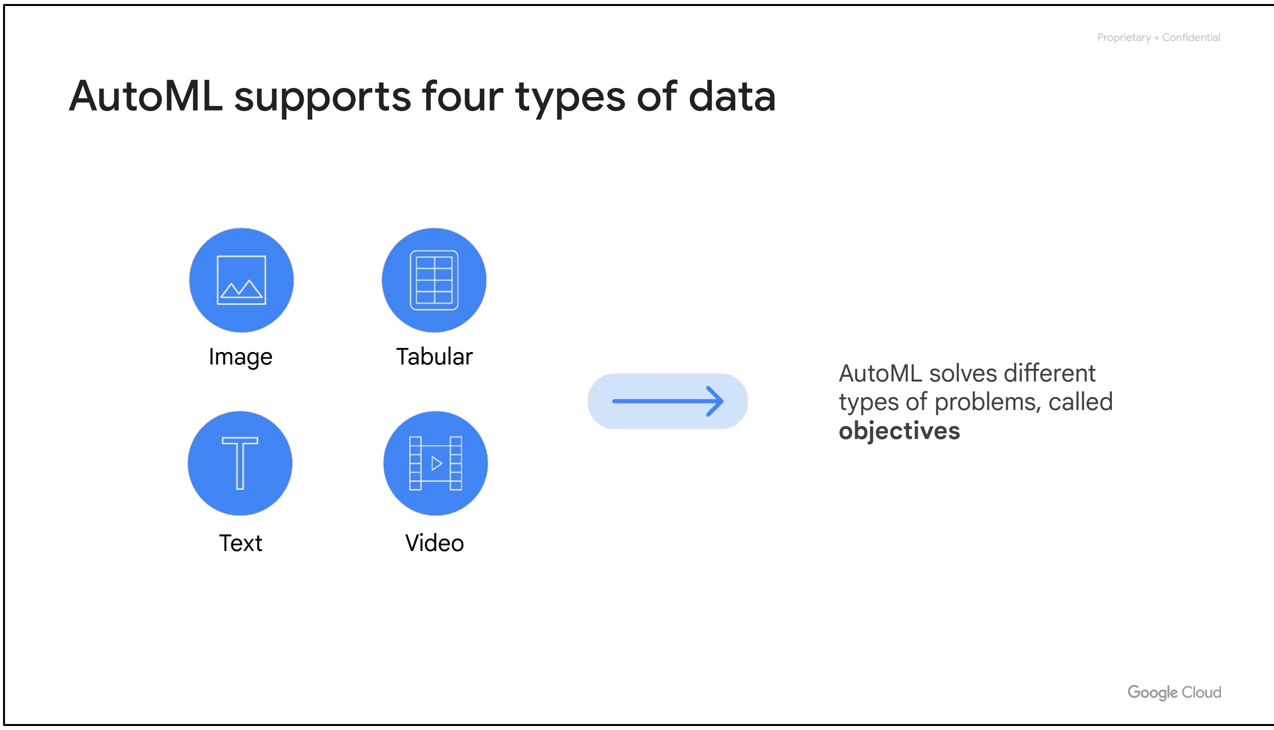
AutoML is powered by the latest machine-learning research, so although a model performs training, the AutoML platform actually trains and evaluates multiple models and compares them to each other. This neural architecture search produces an ensemble of ML models and chooses the best one.

Leveraging these technologies has produced a tool that can significantly benefit data scientists.



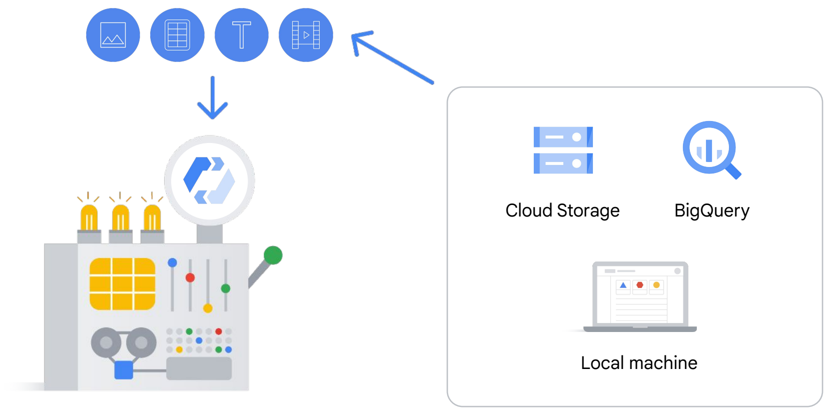
One of the biggest benefits is that it’s a **no-code solution**.That means it can train high-quality custom machine learning models with minimal effort and requires little machine learning expertise.

This allows data scientists to focus their time on tasks like defining business problems or evaluating and improving model results.
Others might find AutoML useful as a tool to quickly prototype models and explore new datasets before investing in development. This might mean using it to identify the best features in a dataset, for example.

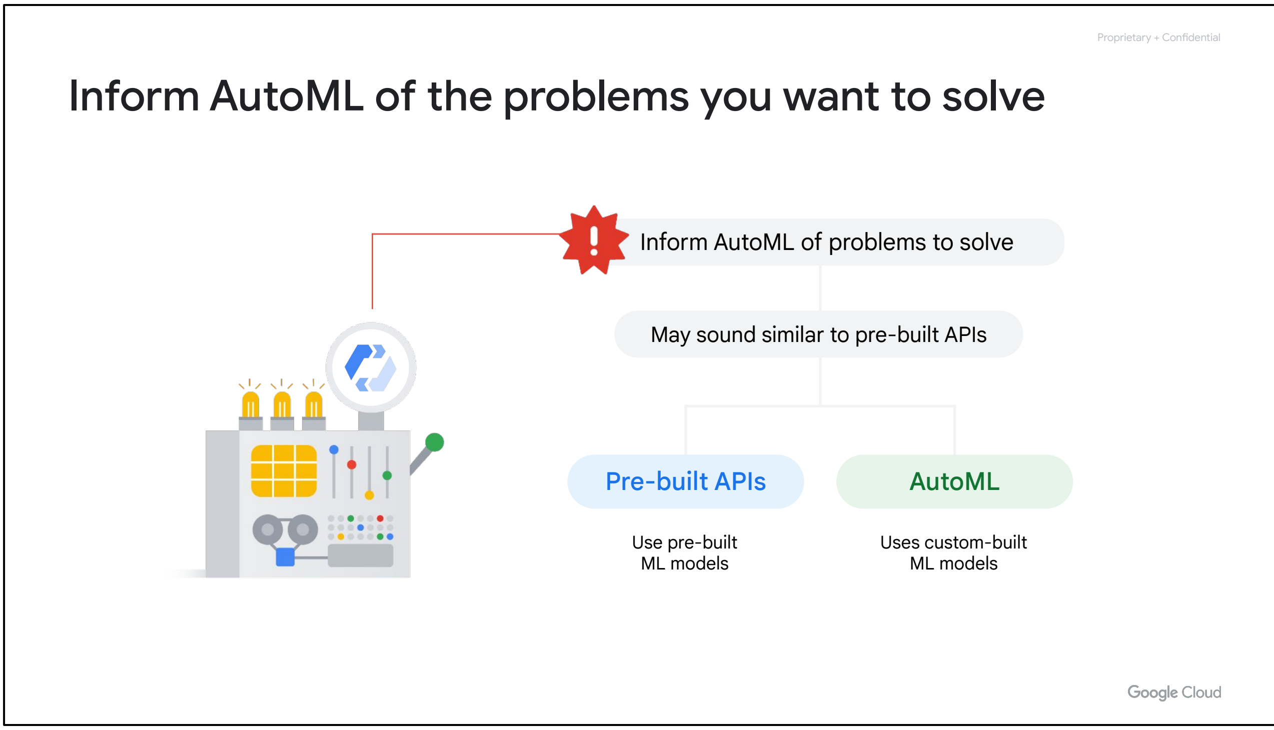


So, how does AutoML work exactly? AutoML supports four types of data: **image**, **tabular**, **text**, and **video**. For each data type, AutoML solves different types of problems, called **objectives.**

To start, upload your data into AutoML

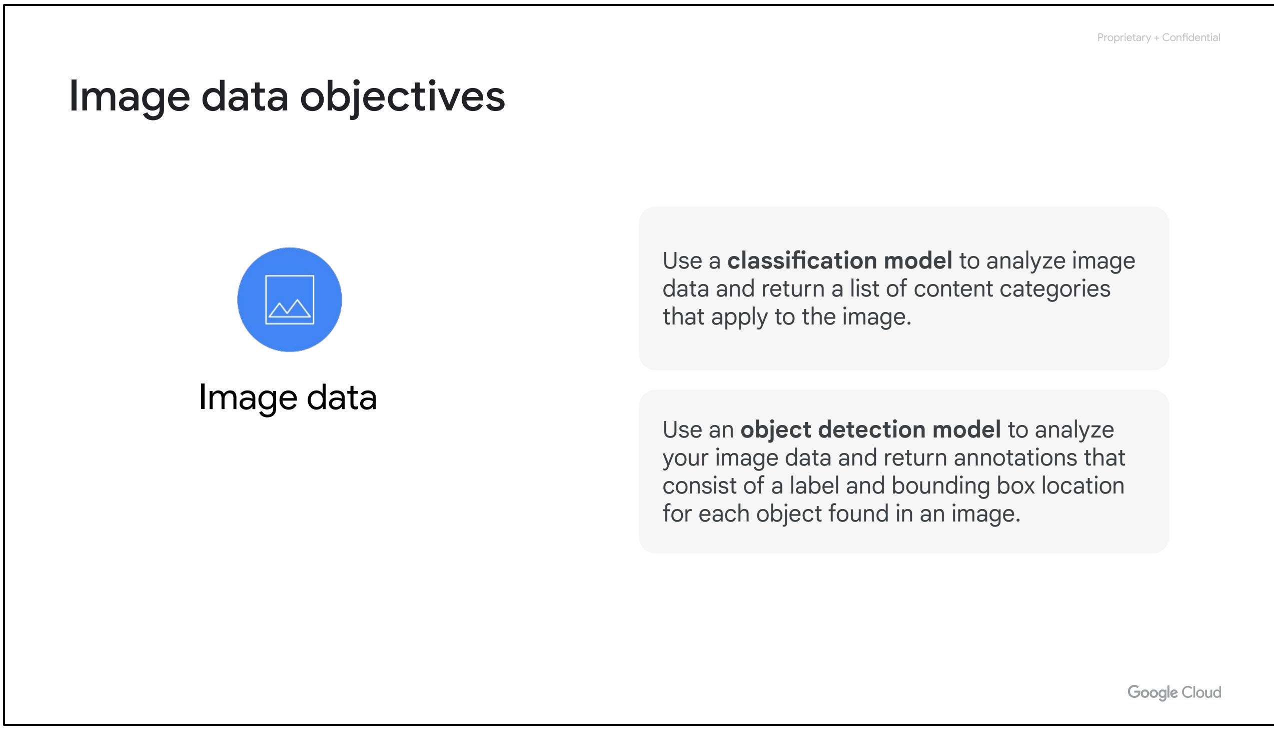


To get started, upload your data into AutoML. It can come from Cloud Storage, BigQuery, or even your local machine.



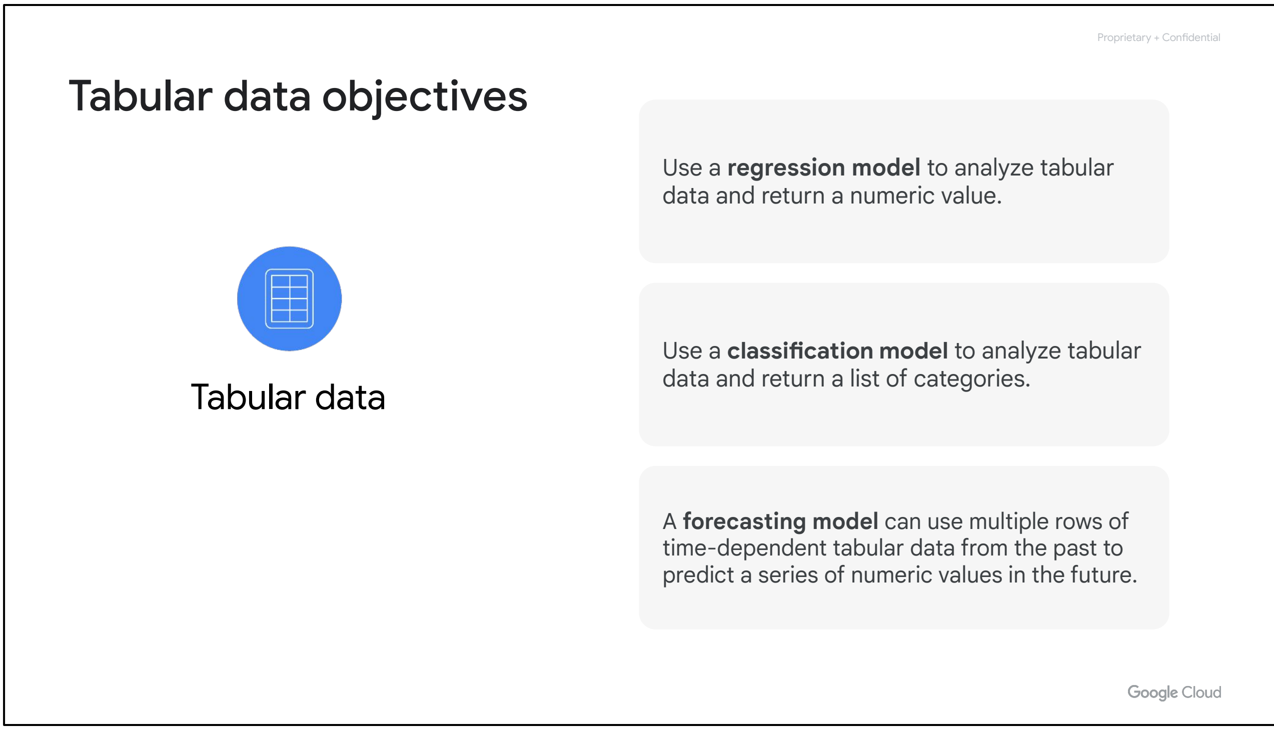
From there, inform AutoML of the problems you want to solve.

Some problems may sound similar to those mentioned in pre-built APIs. However the major difference is that pre-built APIs use pre-built machine learning models, while AutoML uses custom-built models. In AutoML, you use your own data to train the machine learning model and then apply the trained model to predict your goal. While in pre-built APIs, the models are already trained with Google's datasets. You take the advantage of the training results to predict your data.



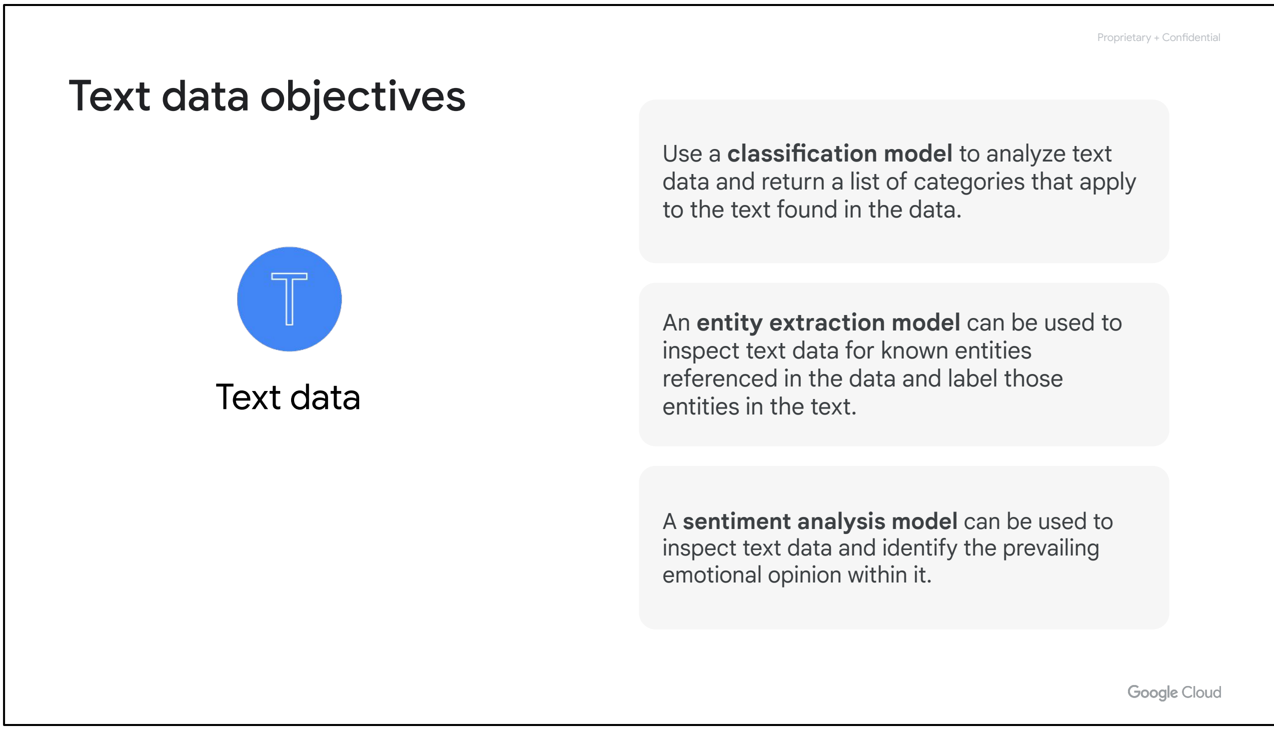
For **image data**:

- You can use a **classification model** to analyze image data and return a list of content categories that apply to the image. For example, you could train a model that classifies images as containing a dog or not containing a dog, or you could train a model to classify images of dogs by breed.
- You can also use an **object detection model** to analyze your image data and return annotations that consist of a label and bounding box location for each object found in an image. For example, you could train a model to find the location of the dogs in image data.



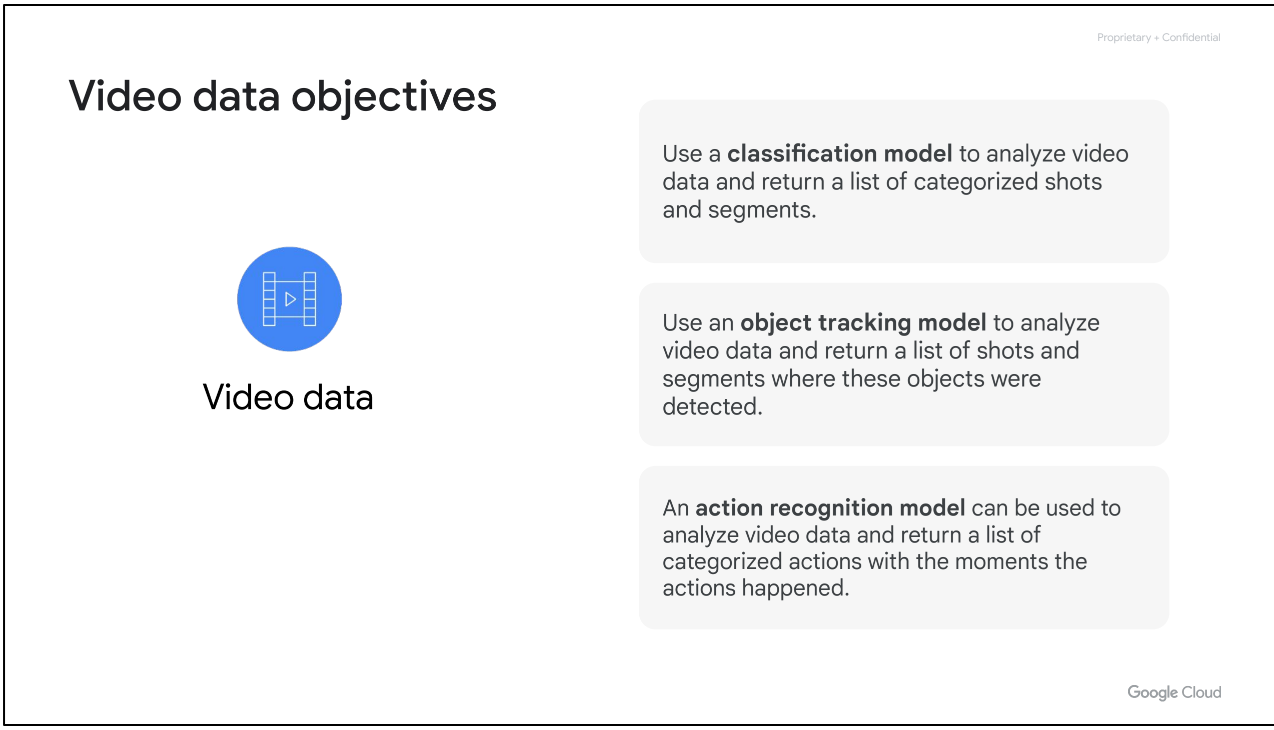
For **tabular data**:

- You can use a **regression model** to analyze tabular data and return a numeric value. For example, you could train a model to estimate a house's value or rental price based on a set of factors such as location, size of the house, and number of bedrooms.
- You can use a **classification model** to analyze tabular data and return a list of categories. For example, you could train a model to classify different types of land into high, median, and low potentials for commercial real estate.
- And a **forecasting model** can use multiple rows of time-dependent tabular data from the past to predict a series of numeric values in the future. For example, you could use the historical *plus* the economic data to predict what the housing market will look like in the next five years.



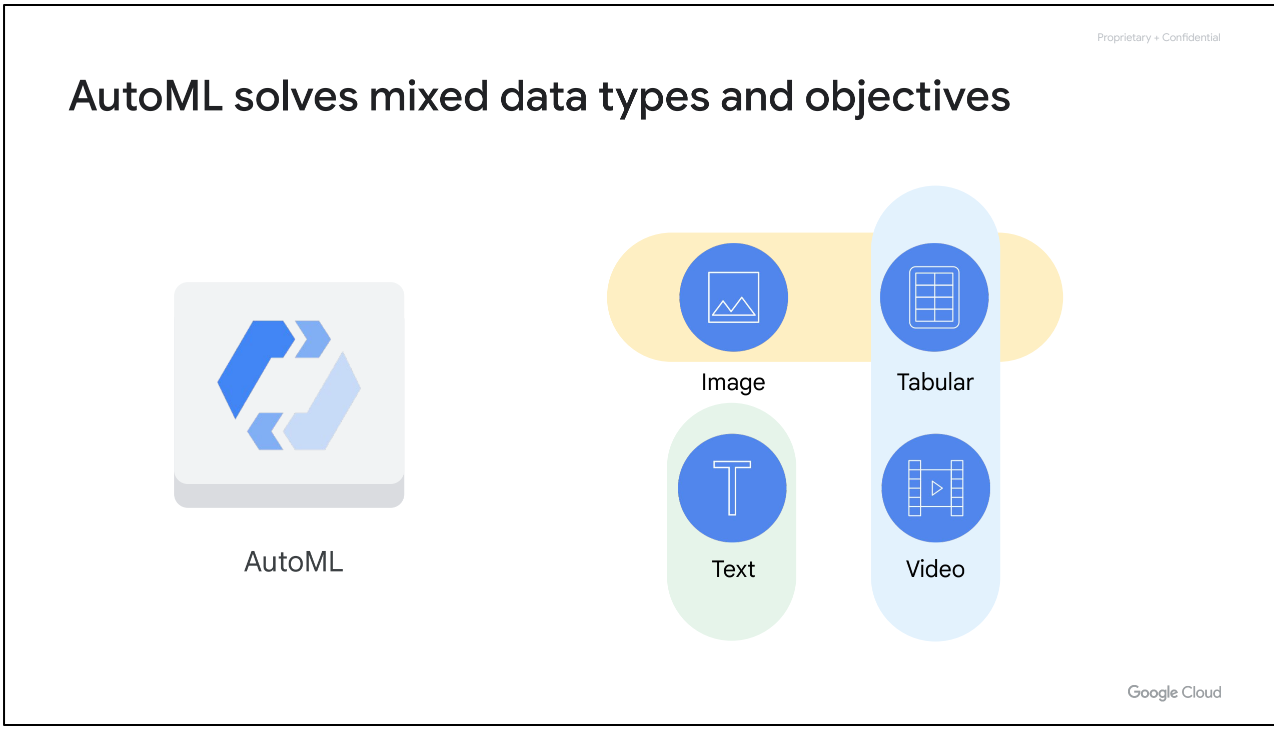
For **text data**:

- You can use a **classification model** to analyze text data and return a list of categories that apply to the text found in the data. For example, you can classify customer questions and comments to different categories and then redirect them to corresponding departments.
- An **entity extraction model** can be used to inspect text data for known entities referenced in the data and label those entities in the text. For example, you can label a social media post in terms of predefined entities such as time, location, and topic, etc. This can help with online search, similar to the concept of a hashtag, but created by machine.
- And a **sentiment analysis model** can be used to inspect text data and identify the prevailing emotional opinion within it, especially to determine a writer's comment as positive, negative, or neutral.



And finally, for **video data**:

- You can use a **classification model** to analyze video data and return a list of categorized shots and segments. For example, you could train a model that analyzes video data to identify whether the video is of a soccer, baseball, basketball, or football game.
- You can use an **object tracking model** to analyze video data and return a list of shots and segments where these objects were detected. For example, you could train a model that analyzes video data from soccer games to identify and track the ball.
- And an **action recognition model** can be used to analyze video data and return a list of categorized actions with the moments the actions happened. For example, you could train a model that analyzes video data to identify the action moments involving a soccer goal, a golf swing, a touchdown, or a high five.

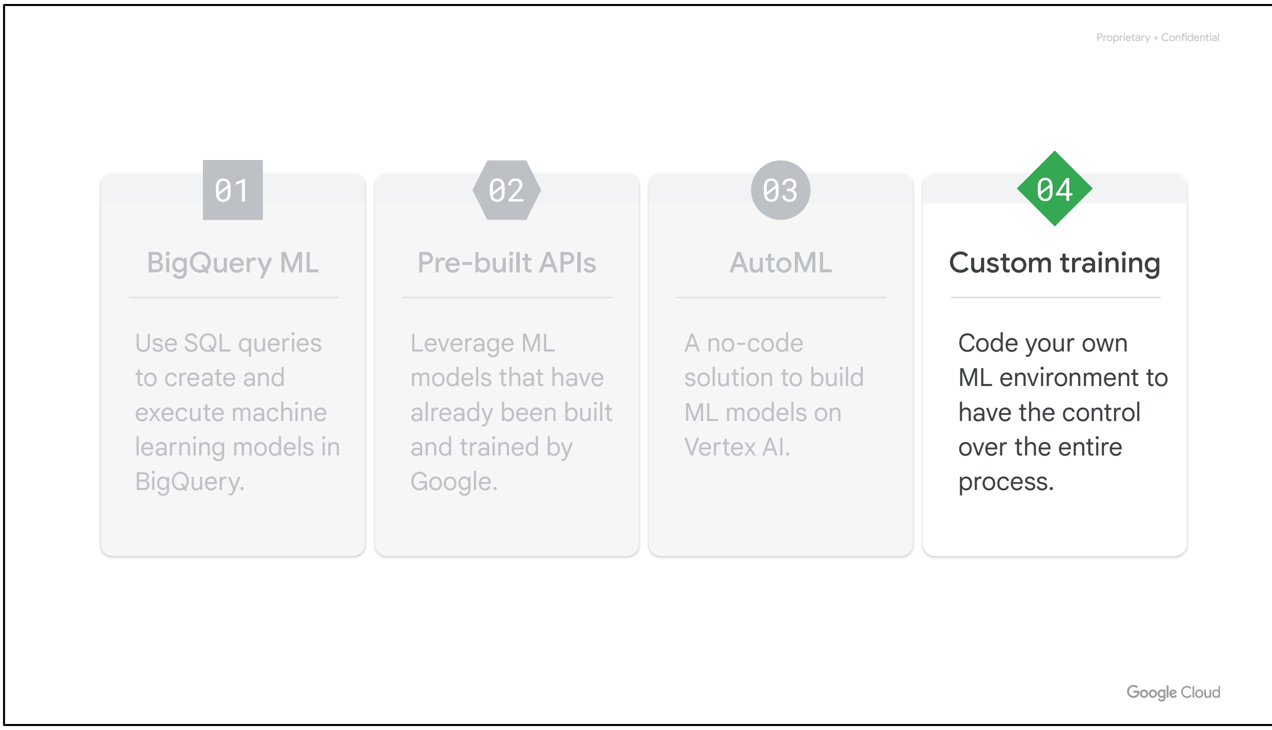


In reality, you may not be restricted to just one data type and one objective but rather need to combine different objectives to solve a business problem.

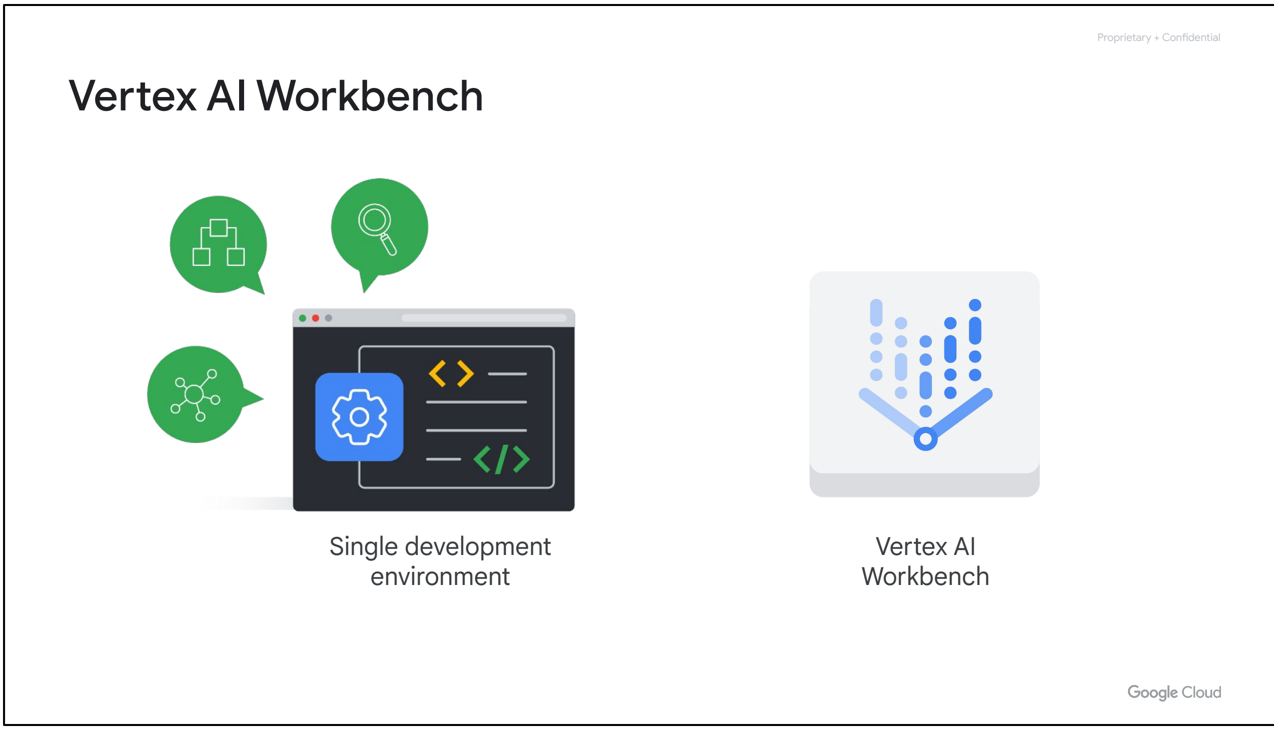
AutoML is a powerful tool that can help across these different data types and objectives.

05

Custom training



We've explored the options Google Cloud provides to build machine learning models using BigQuery ML, pre-built APIs, and AutoML. Now let's take a look at the last option, **custom training**, which allows you to code your own ML environment to have the control over the entire ML development process starting from data preparation to model deployment.



If you want to code your machine learning model, you can use this option by building a custom training solution with **Vertex AI Workbench**.

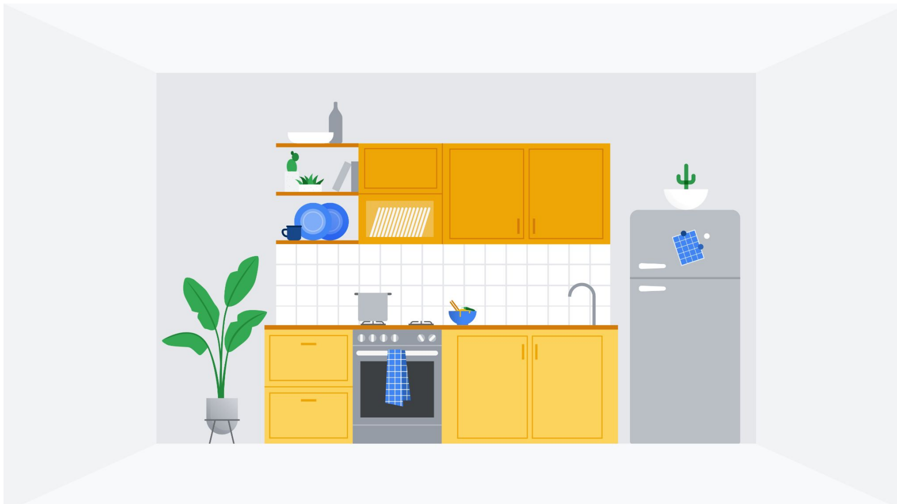
Workbench is a **single development environment** for the entire data science workflow, from exploring, to training, and then deploying a machine learning model with code.

ML training code environment options

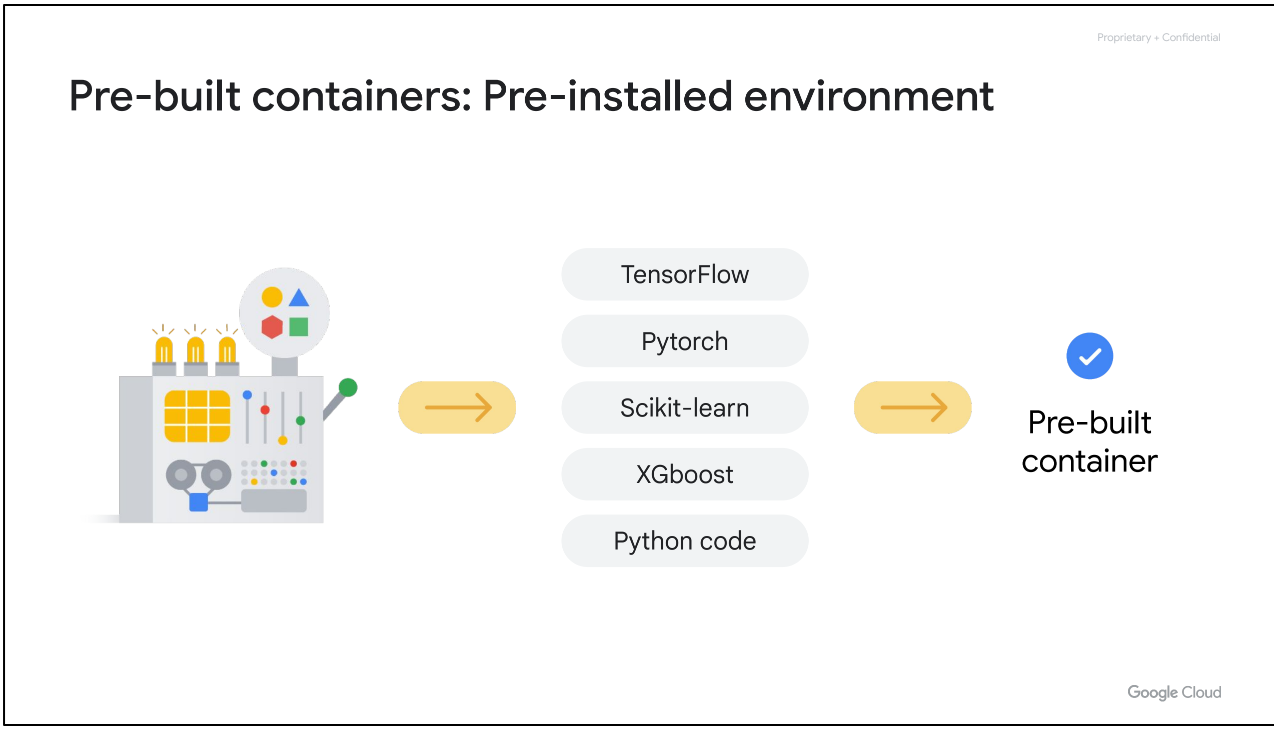
- A pre-built container
- A custom container

Before any coding begins, you need to determine what environment you want your ML training code to use. There are two options: a **pre-built container** or a **custom container**.

A pre-built container is like a fully furnished room

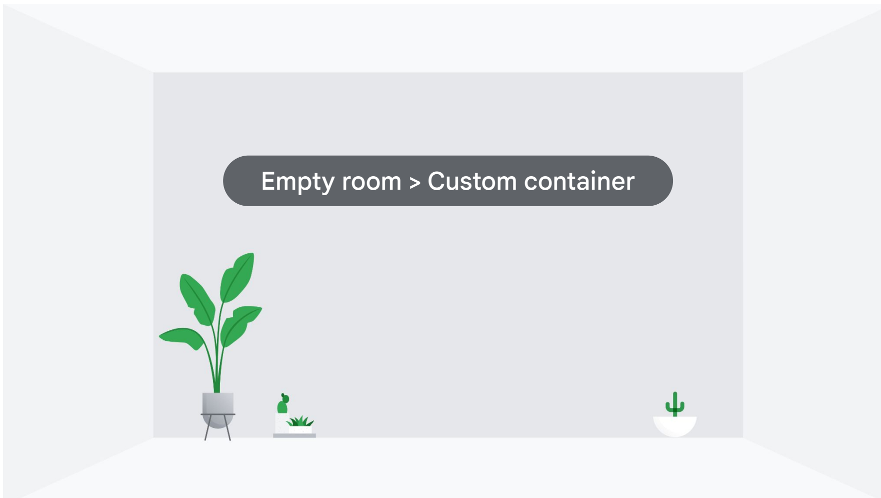


Imagine that a container is a kitchen. A pre-built container would represent a fully furnished room with cabinets and appliances (which represent the dependencies) that includes all the cookware (which represents the libraries) you need to make a meal.



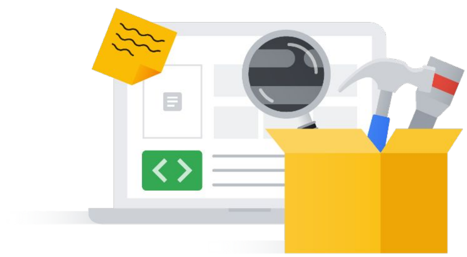
So, if your ML training needs a platform like TensorFlow, Pytorch, Scikit-learn, or XGboost, and Python code to work with the platform, a pre-built container is probably your best solution.

A custom container is like an empty room



A custom container, alternatively, is like an empty room with no cabinets, appliances, or cookware.

Custom container: Define your own environment

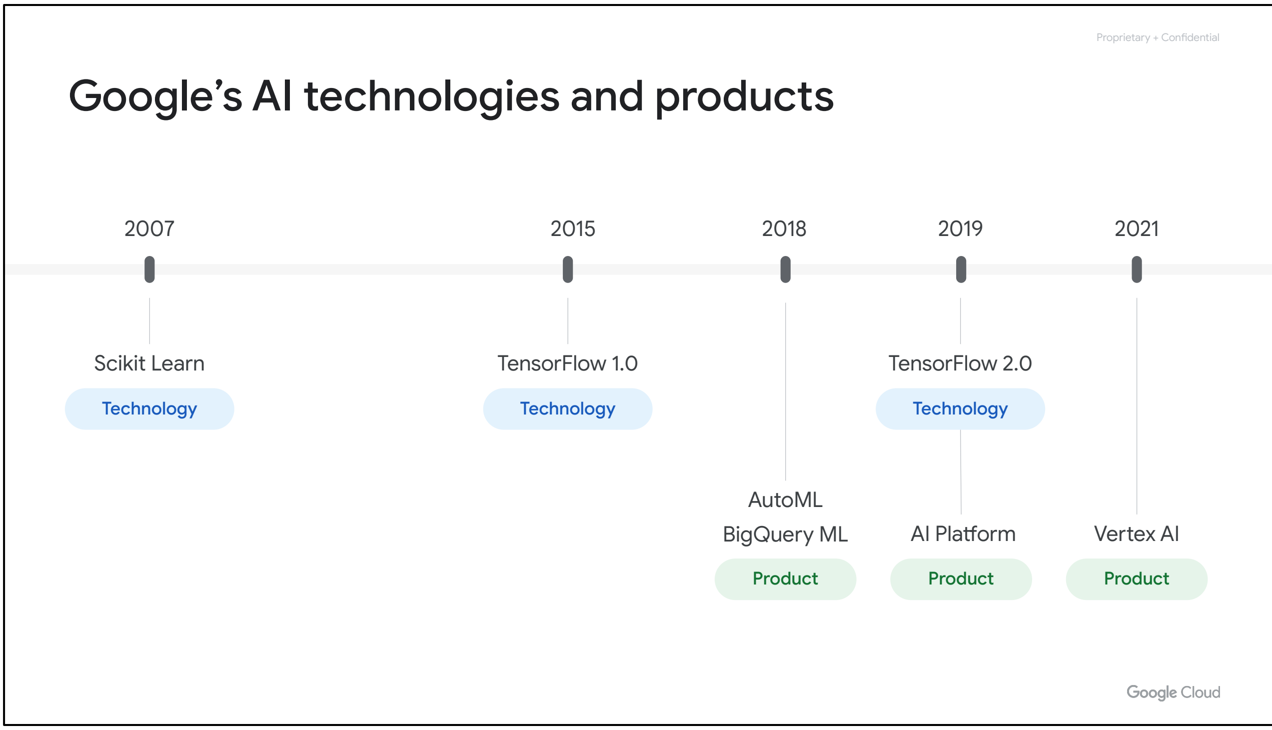


Custom container

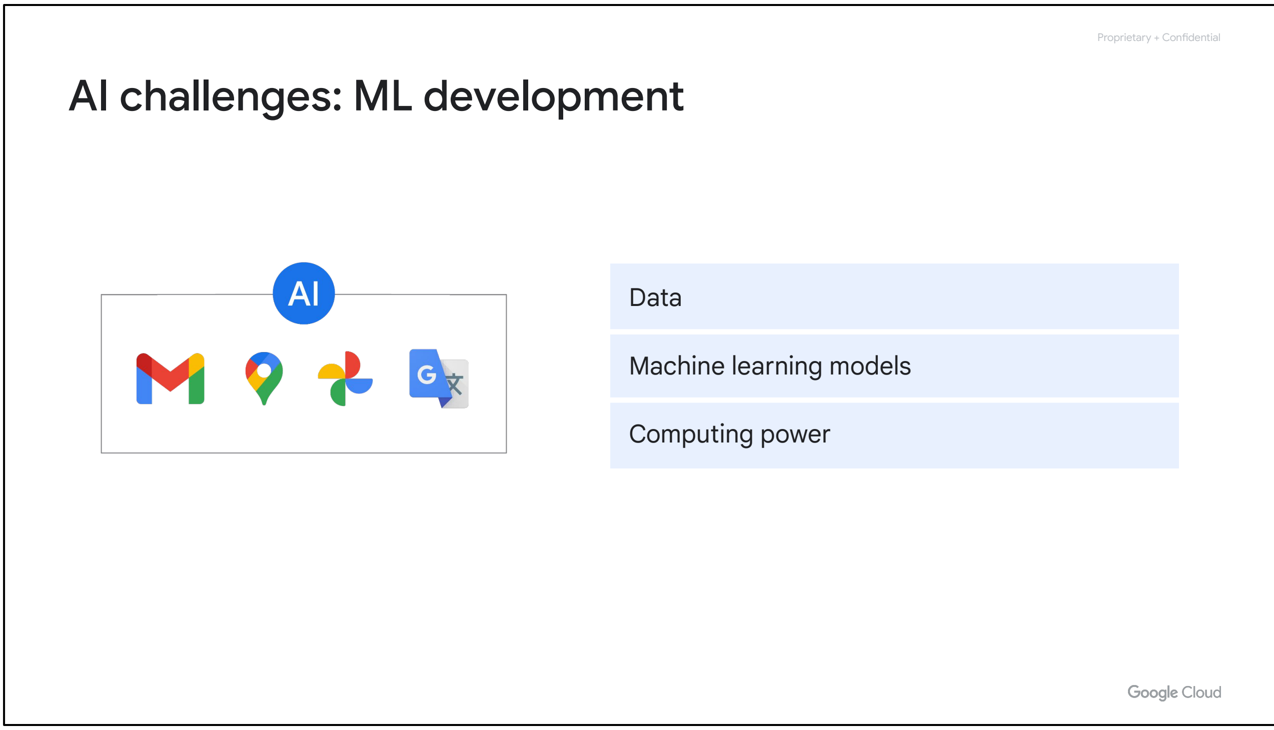
You define the exact tools that you need to complete the job.

06

Vertex AI

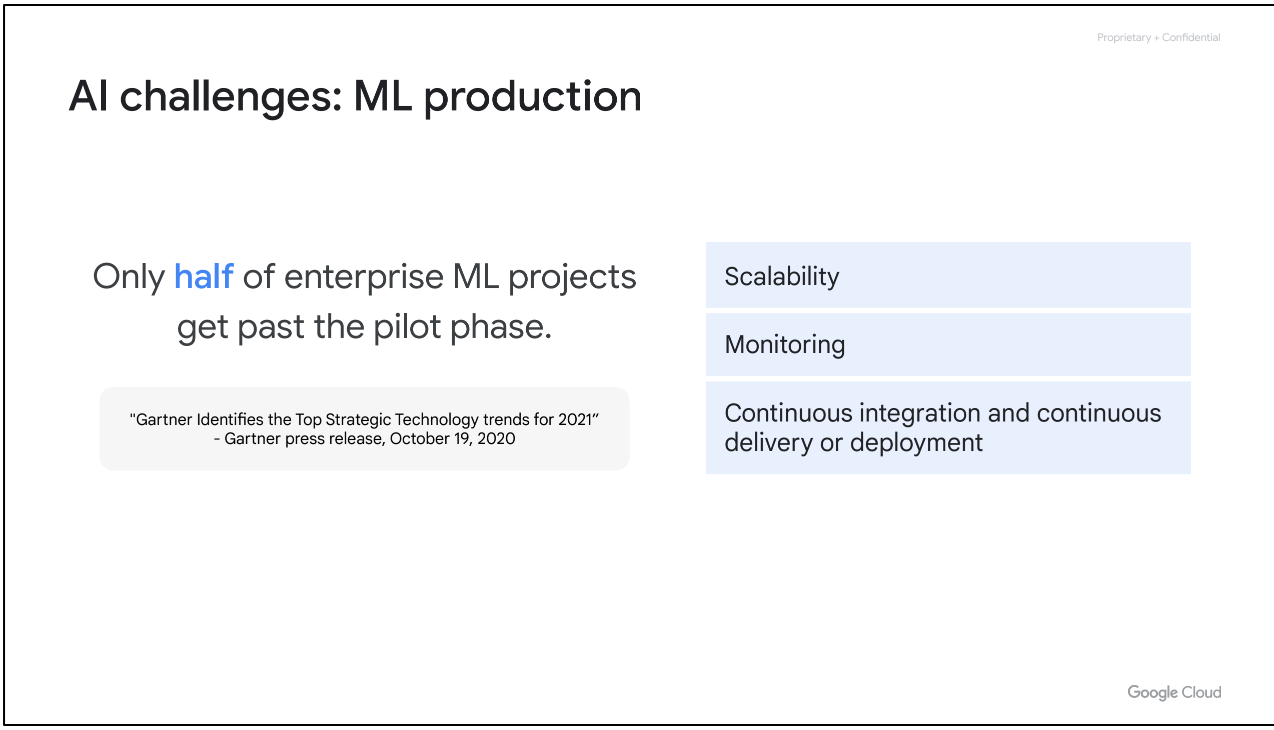


For years now, Google has invested time and resources into developing big data and AI. Google had developed key technologies and products, from **scikit-learn** as a Google summer coding project back in 2007 to Vertex AI today.



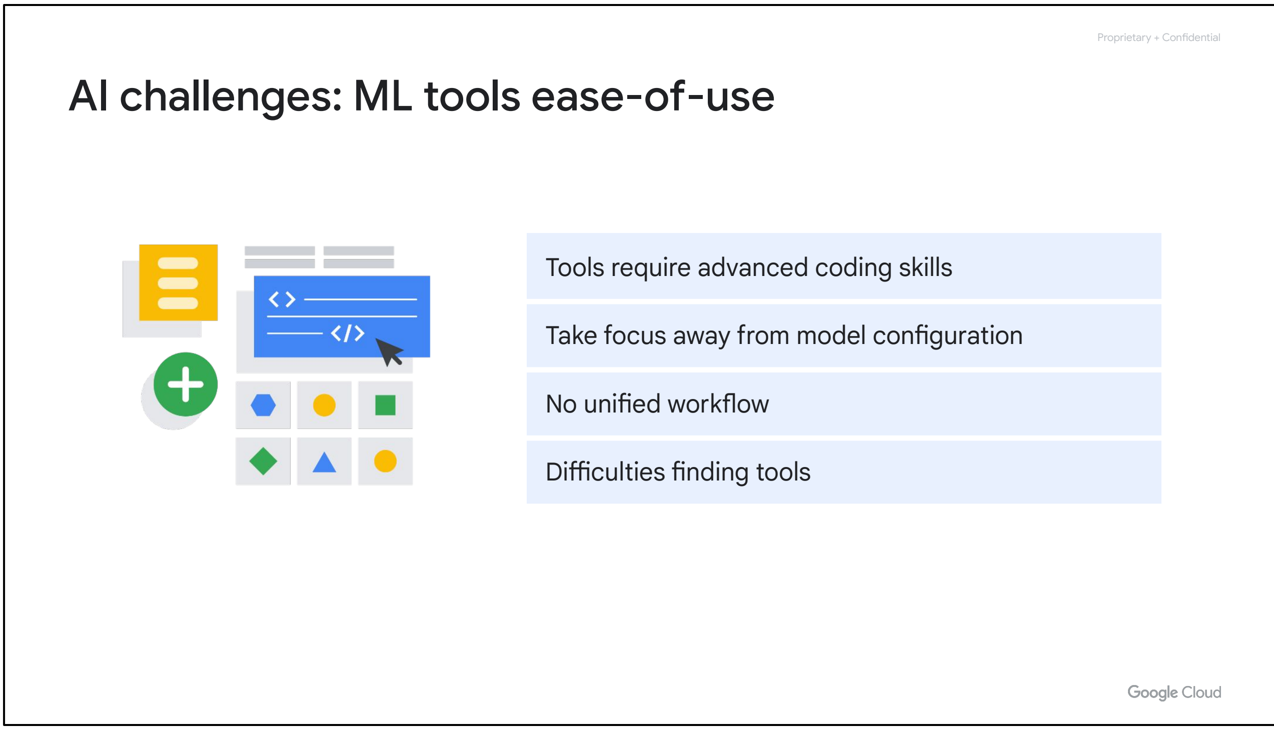
As an AI-first company, Google has applied AI technologies to many of its products and services, like Gmail, Google Maps, Google Photos, and Google Translate, just to name a few. But developing these technologies doesn't come without challenges, especially when it involves **developing machine learning models** and **putting them into production**.

Some **traditional challenges** include determining how to handle large quantities of data, determining the right machine learning model to train the data, and harnessing the required amount of computing power.

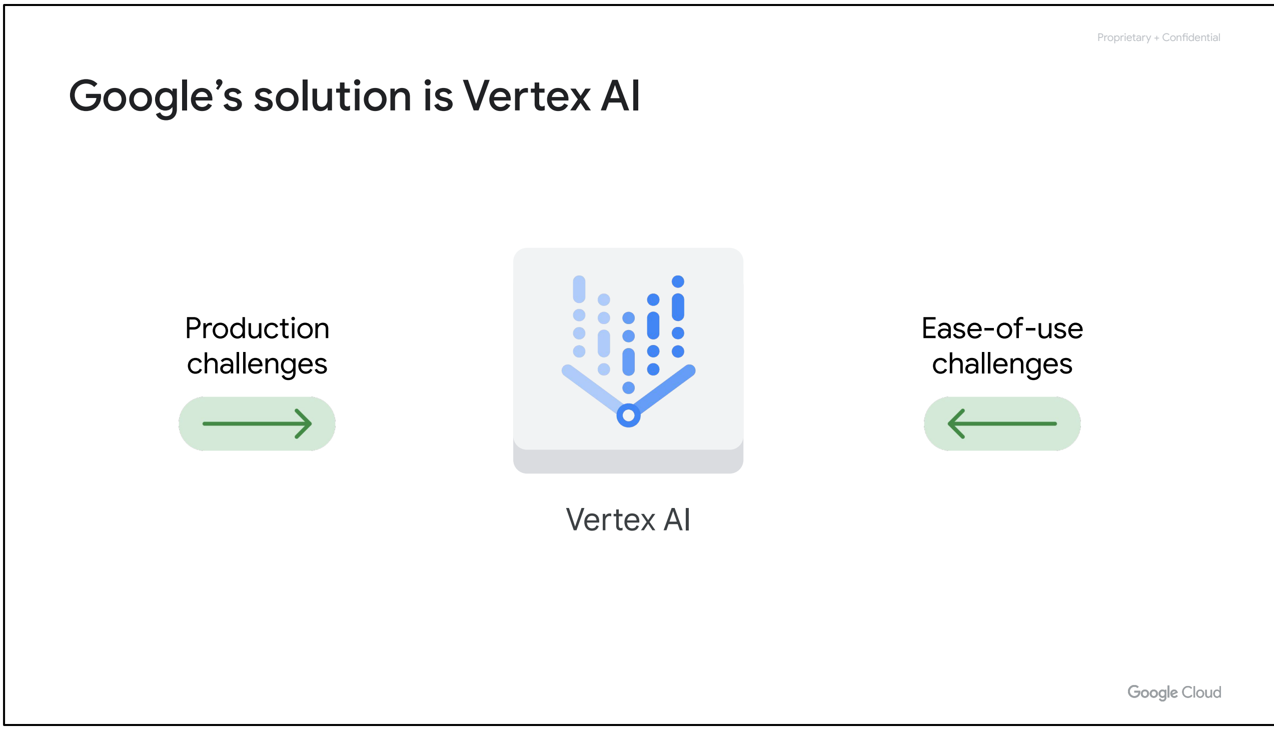


Then there are challenges around getting ML models into production. **Production challenges** require scalability, monitoring, and continuous integration and continuous delivery or deployment.

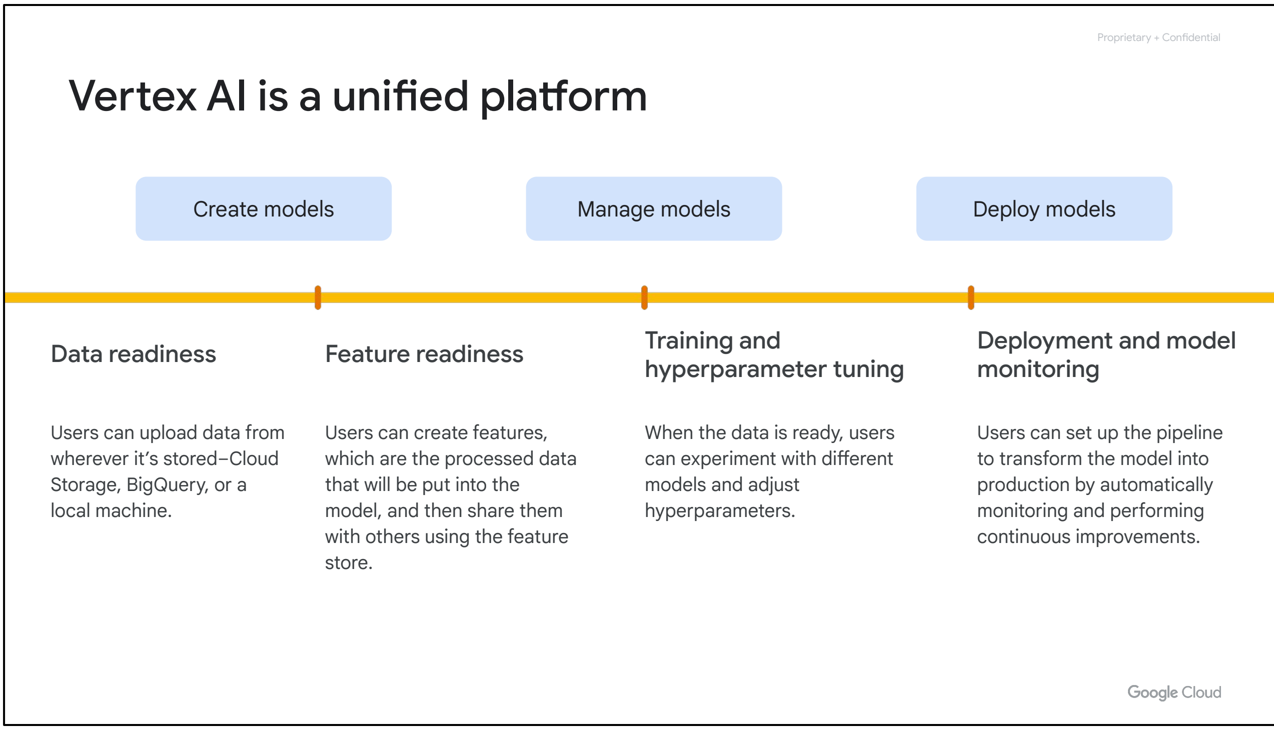
In fact, according to Gartner, only **half** of enterprise ML projects get past the pilot phase.



And finally, there are **ease-of-use challenges**. Many tools on the market require advanced coding skills, which can take a data scientist's focus away from model configuration. And without a unified workflow, data scientists often have difficulties finding tools.

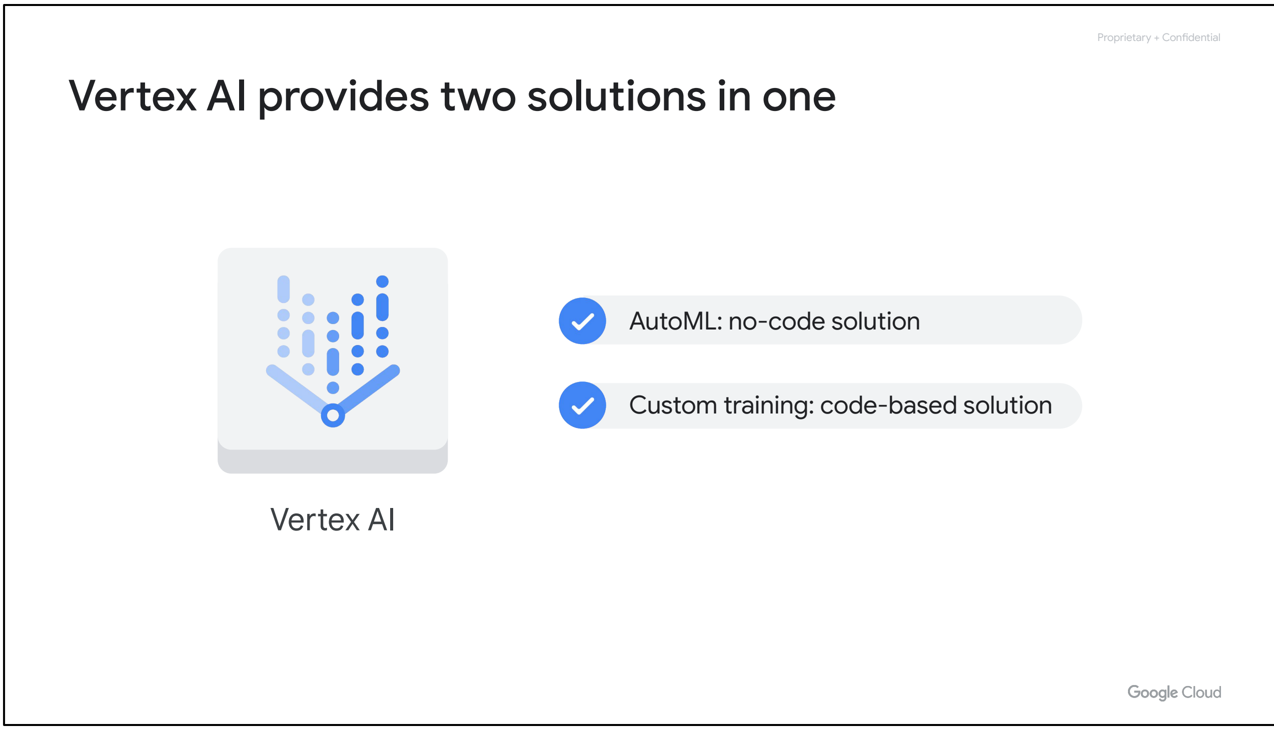


Google's solution to many of the **production** and **ease-of-use challenges** is **Vertex AI**, a **unified** platform that brings all the components of the machine learning ecosystem and workflow together.

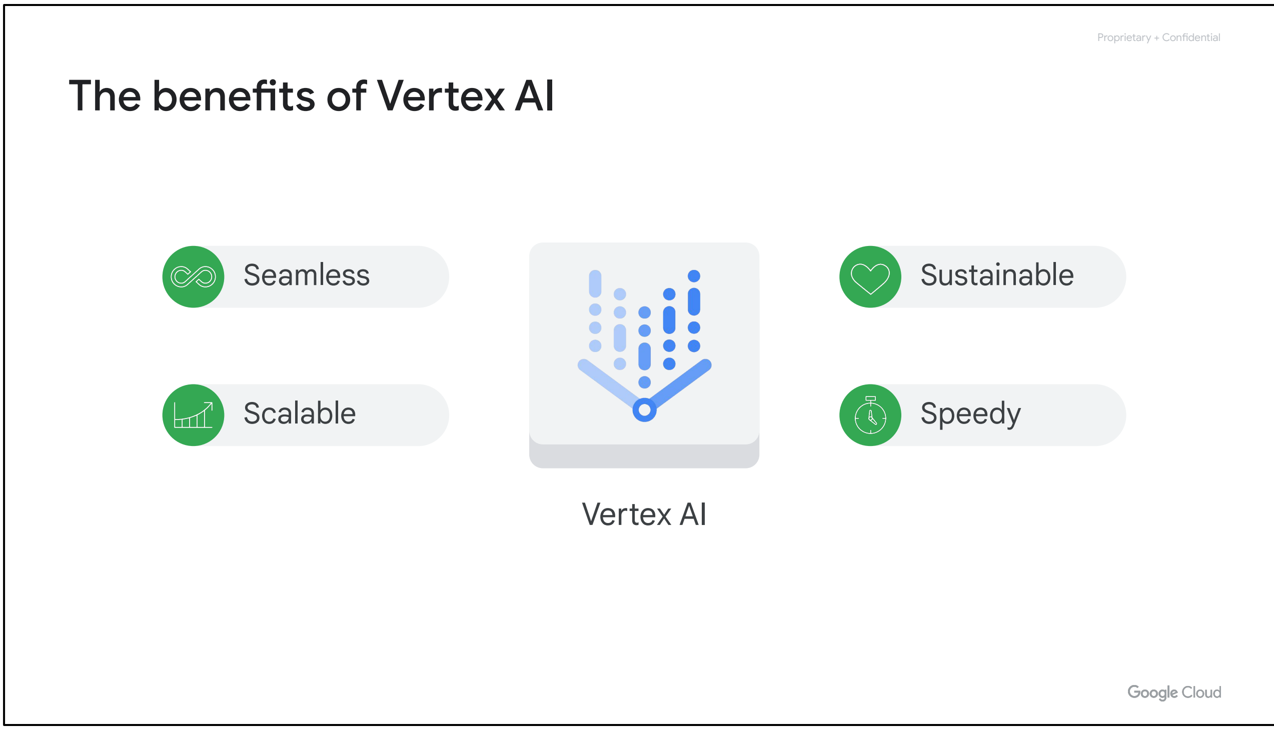


So, what exactly does a unified platform mean? In the case of Vertex AI, it means having one digital experience to create, manage, and deploy models over time, and at scale. For example,

- **During the data readiness** stage, users can upload data from wherever it's stored– Cloud Storage, BigQuery, or a local machine.
- Then, **during the feature readiness stage**, users can create features, which are the processed data that will be put into the model, and then share them with others using the feature store.
- After that, it's time for **Training and Hyperparameter tuning**. This means that when the data is ready, users can experiment with different models and adjust hyperparameters.
- And finally, during **deployment and model monitoring**, users can set up the **pipeline** to transform the model into production by automatically monitoring and performing continuous improvements.



And to refer back to the different options we explored earlier, Vertex AI allows users to build machine learning models with either **AutoML**, a code-less solution or **Custom Training**, a code-based solution. AutoML is easy to use and lets data scientists spend more time turning business problems into ML solutions, while custom training enables data scientists to have full control over the development environment and process.

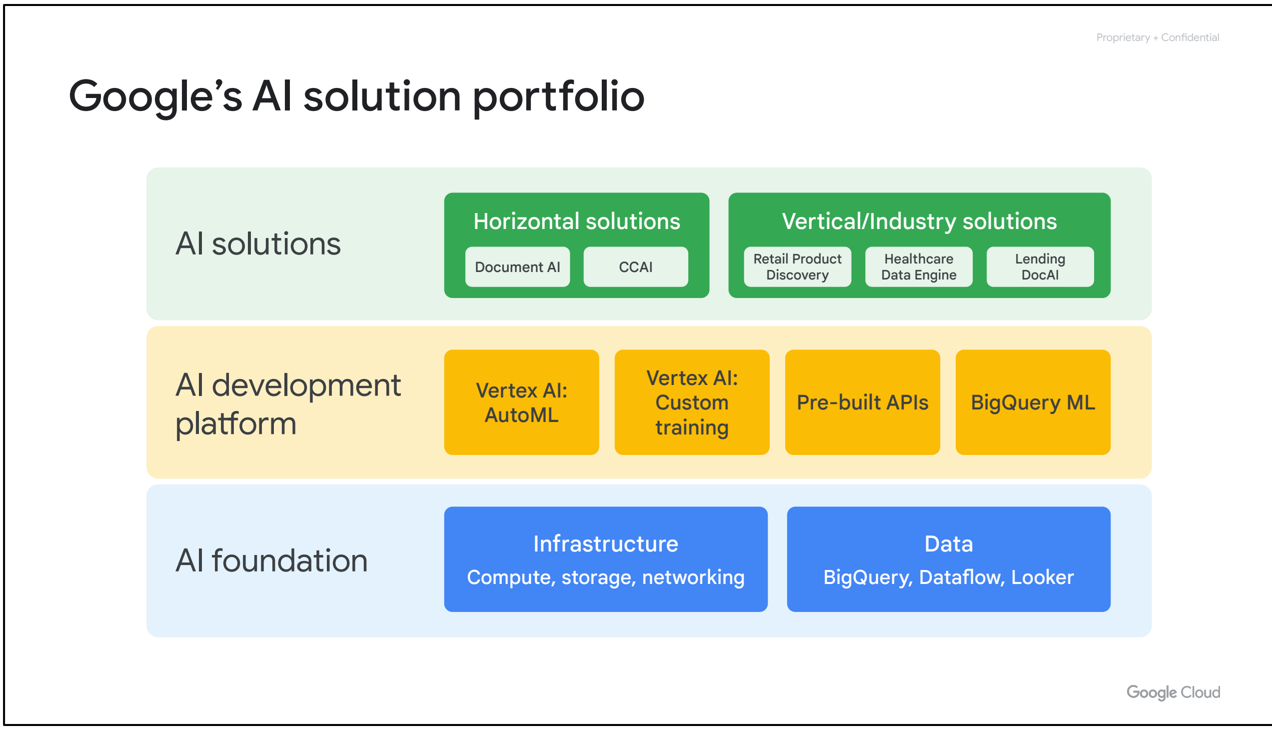


Being able to perform such a wide range of tasks in one unified platform has many benefits. This can be summarized with four Ss:

- It's **seamless**. Vertex AI provides a smooth user experience from uploading and preparing data all the way to model training and production.
- It's **scalable**. The Machine Learning Operations (MLOps) provided by Vertex AI helps to monitor and manage the ML production and therefore scale the storage and computing power automatically.
- It's **sustainable**. All of the artifacts and features created using Vertex AI can be reused and shared.
- And it's **speedy**. Vertex AI produces models that have [80% fewer lines of code] than competitors.

07

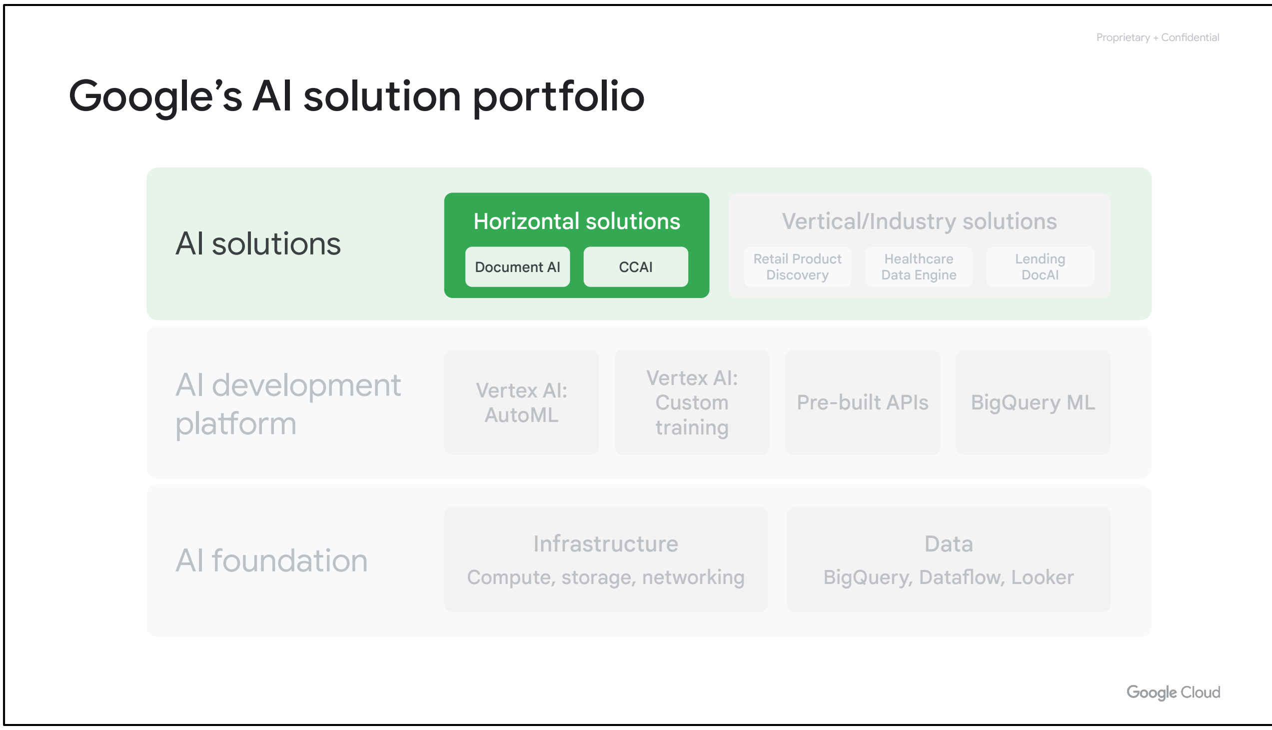
AI solutions



Now that you've explored the four different options available to create machine learning models with Google Cloud, let's take a few minutes to explore Google Cloud's artificial intelligence solution portfolio.

It can be visualized with three layers.

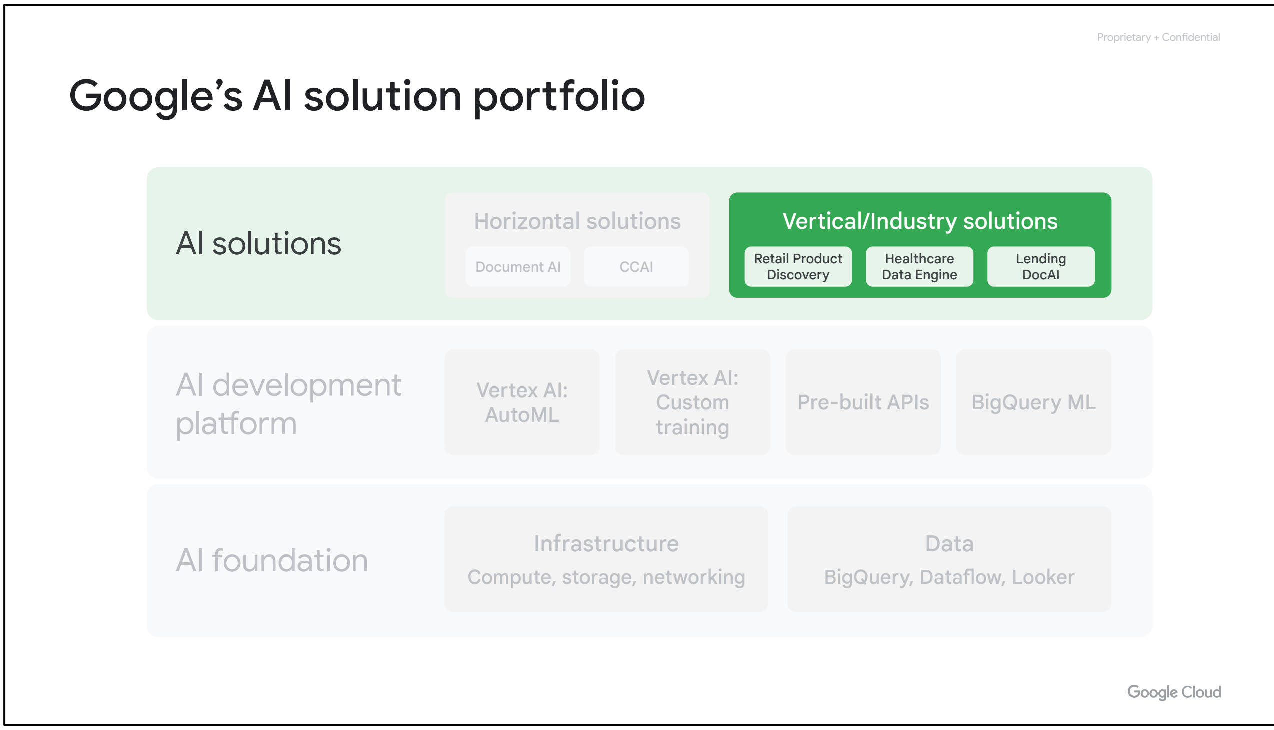
- The bottom layer is the **AI foundation**, and includes the Google Cloud infrastructure and data.
- The middle layer represents the **AI development platform**, which includes the four ML options you just learned about: AutoML and custom training, which are offered through Vertex AI, and pre-built APIs and BigQuery ML.
- The top layer represents **AI solutions**, for which there are two groups, horizontal solutions and industry solutions.



Horizontal solutions usually apply to *any* industry that would like to solve the same problem.

Examples include Document AI and CCAI.

- Document AI uses computer vision and optical character recognition, along with natural language processing, to create pretrained models to extract information from documents. The goal is to increase the speed and accuracy of document processing to help organizations make better decisions faster, while reducing costs.
- Another example of a horizontal solution is Contact Center AI, or CCAI. The goal of CCAI is to improve customer service in contact centers through the use of artificial intelligence. It can help automate simple interactions, assist human agents, unlock caller insights, and provide information to answer customer questions.



And the second group is vertical, or **industry solutions**. These represent solutions that are relevant to specific industries.

Examples include:

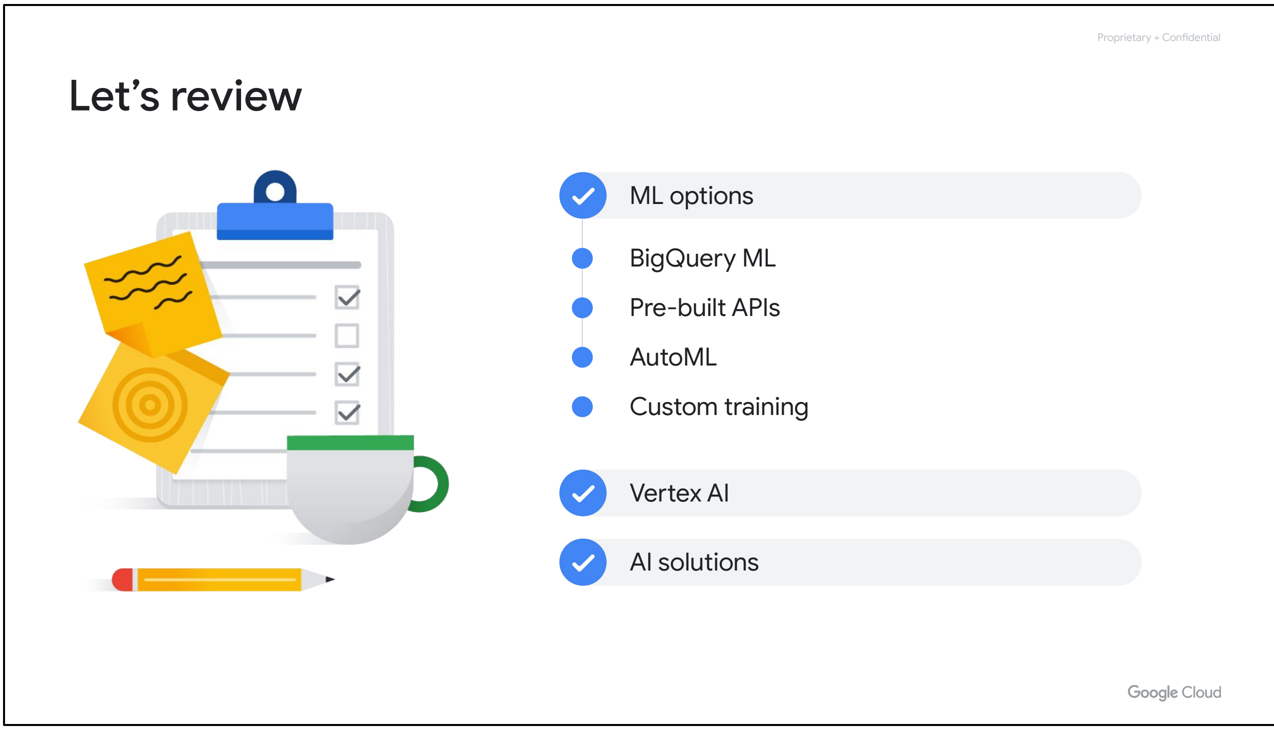
- Retail Product Discovery, which gives retailers the ability to provide Google-quality search and recommendations on their own digital properties, helping to increase conversions and reduce search abandonment,
- Google Cloud Healthcare Data Engine, which generates healthcare insights and analytics with one end-to-end solution, and
- Lending DocAI, which aims to transform the home loan experience for borrowers and lenders by automating mortgage document processing.

cloud.google.com/solutions/ai

You can learn more about Google Cloud's growing list of AI solutions at cloud.google.com/solutions/ai.

08

Summary

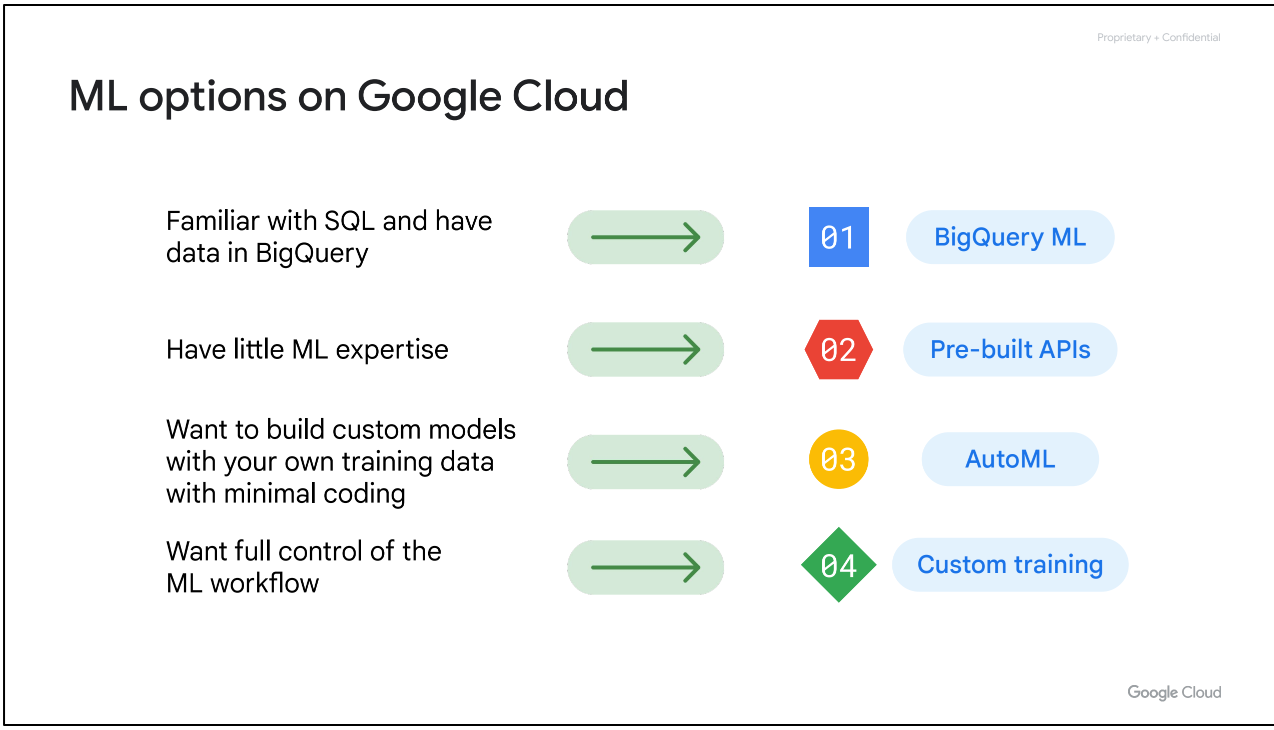


We've covered a lot of information in this module of the course. Let's do a quick recap.

To start, we explored Google's history as an AI-first company. From there, we looked at the four options Google Cloud offers to build machine learning models, BigQuery ML, pre-built APIs, AutoML, and Custom Training. We focused on the latter three in this module.

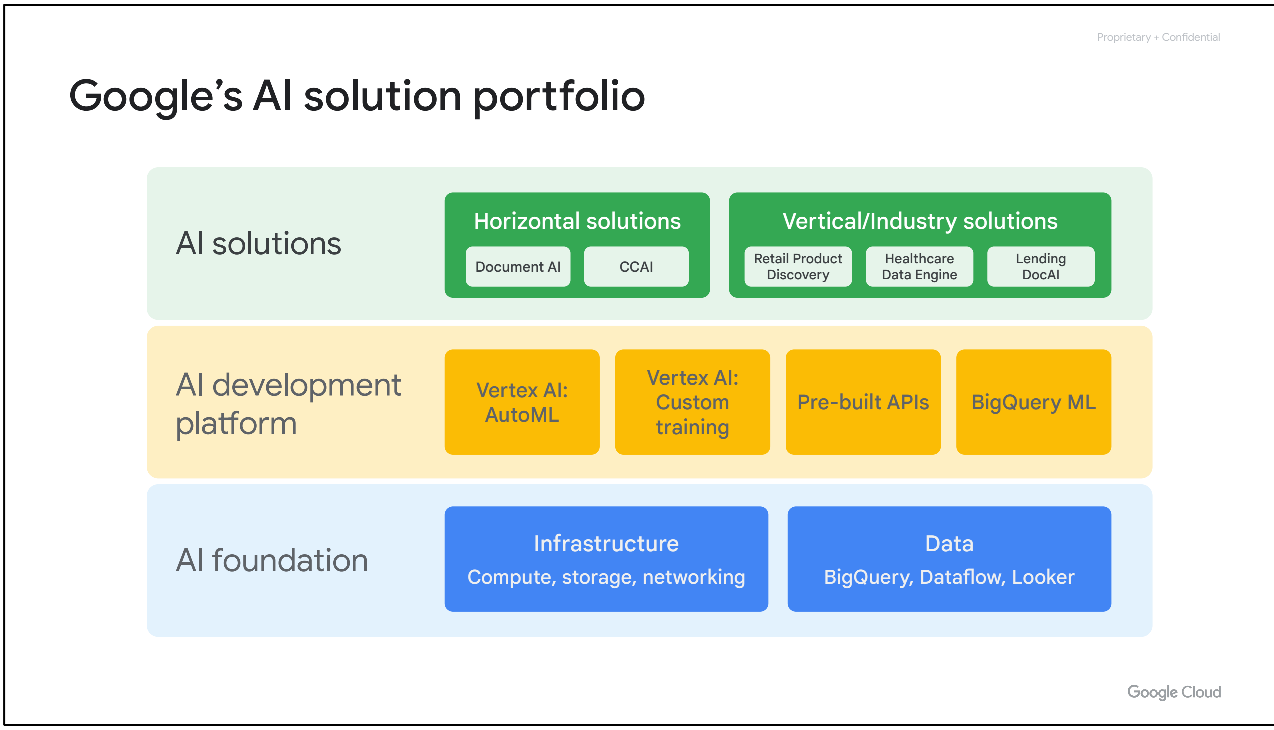
Next, we introduced Vertex AI, a tool that combines the functionality of AutoML, which is codeless, and custom training, which is code-based, to solve production and ease-of-use problems.

And finally, we introduced Google Cloud AI solutions.



You'll recall that selecting the best ML option will depend on your business needs and ML expertise.

- If your data engineers, data scientists, and data analysts are familiar with SQL and already have your data in BigQuery, BigQuery ML lets you develop SQL-based models.
- If your business users or developers have little ML experience, using pre-built APIs is likely the best choice. Pre-built APIs address common perceptual tasks such as vision, video, and natural language. They are ready to use without any ML expertise or model development effort.
- If your developers and data scientists want to build custom models with your own training data while spending minimal time coding, then AutoML is your choice. AutoML provides a code-less solution to enable you to focus on business problems instead of the underlying model architecture and ML provisioning.
- If your ML engineers and data scientists want full control of ML workflow, Vertex AI custom training lets you train and serve custom models with code on Vertex Workbench.



The Google AI solutions are built on top of the four ML development options to meet both horizontal and vertical market needs.