Google Cloud

Partner Certification Academy

Professional Machine Learning Engineer

pls-academy-pmle-student-slides-8-2403

The information in this presentation is classified:

Google confidential & proprietary

⚠️ This presentation is shared with you under NDA.

- Do **not** record or take screenshots of this presentation.
- Do **not** share or otherwise distribute the information in this presentation with anyone **inside** or **outside** of your organization.

Thank you!

Source Materials

Some of this program's content has been sourced from the following resources:

- Google Cloud certification site
- Google Cloud documentation
- Google Cloud console
- Google Cloud courses and workshops
- Google Cloud white papers
- Google Cloud Blog
- Google Cloud YouTube channel
- Google Cloud samples
- Google codelabs
- Google Cloud partner-exclusive resources

Google Cloud

Google Cloud Skills Boost for Partners

- Professional Machine Learning Engineer Certification
- Cloud Skills Boost for Partners Professional Machine Learning Engineer Learning Path
- Partner Learning Services Instructor-Led PMLE Curriculum

Google Cloud

Partner Advantage

- Best practices for implementing machine learning on Google Cloud
- Artificial Intelligence
- End-to-End MLOps Go-to-Market Kit

Session Logistics

- When you have a question, please:
  - Click the Raise hand button in Google Meet.
  - Or add your question to the Q&A section of Google Meet.
  - Please note that answers may be deferred until the end of the session.

- These slides are available in the Student Lecture section of your Qwiklabs classroom.

- The session is **not recorded.**

- Google Meet does not have persistent chat.
  - If you get disconnected, you will lose the chat history.
  - Please copy any important URLs to a local text file as they appear in the chat.

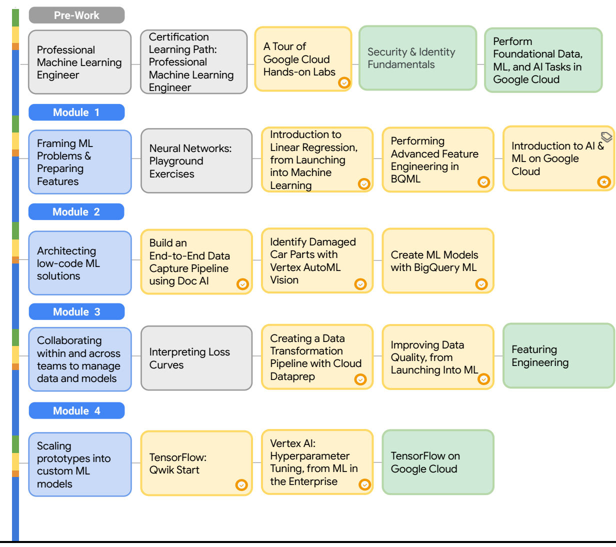
Google Cloud Partner Learning Programs

- Partner Certification Academy
- Partner Delivery Readiness Index (DRI)
- Cloud Skills Boost for Partners
- Partner Advantage

PARTNER CERTIFICATION ACADEMY

Professional Machine Learning Engineer

On-demand Course
Hands-on Skill Badge
Hands-on Lab
Resource
Live Virtual Workshop
required to earn exam voucher



A Professional Machine Learning Engineer builds, evaluates, productionizes, and optimizes ML models by using Google Cloud technologies and knowledge of proven models and techniques. The ML Engineer:

- handles large, complex datasets and creates repeatable, reusable code.
- considers responsible AI and fairness throughout the ML model development process, and collaborates closely with other job roles to ensure long-term success of ML-based applications.
- has strong programming skills and experience with data platforms and distributed data processing tools.
- is proficient in the areas of model architecture, data and ML pipeline creation, and metrics interpretation.
- is familiar with foundational concepts of MLOps, application development, infrastructure management, data engineering, and data governance.
- makes ML accessible and enables teams across the organization.

By training, retraining, deploying, scheduling, monitoring, and improving models, the ML Engineer designs and creates scalable, performant solutions.

Recommended candidate:

- Has in-depth experience setting up cloud environments for an organization
- Has experience deploying services and solutions based on business requirements

Google Cloud

Proprietary + Confidential

PARTNER CERTIFICATION ACADEMY

Professional Machine Learning Engineer

On-demand Course
Hands-on Skill Badge
Hands-on Lab
Resource
Live Virtual Workshop
required to earn exam voucher

Module 5

- Serving & scaling models
- Deploy a BigQuery ML Customer Churn Classifier to Vertex AI for Online Predictions
- Vertex AI: Predicting Loan Risk with AutoML
- Google Cloud Solutions II: Data and Machine Learning

Module 6

- Automating and orchestrating ML pipelines
- GCP Pipelines Components List
- Vertex Pipelines: Qwik Start
- Building End to End MLOps Pipelines using Vertex AI
- ML Ops Using Vertex AI

Module 7

- Monitoring ML solutions
- Attack discrimination with smarter machine learning
- VPC Network Peering
- Monitoring a Vertex AI Model, from ML in the Enterprise
- Build and Deploy Machine Learning Solutions on Vertex AI

Module 8

- Additional Material
- Guidelines for developing high quality ML solutions
- Rules of Machine Learning
- Exam Readiness Workshop
- Sample PMLE Questions

Continued Learning

- Production Machine Learning Systems
- Computer Vision Fundamentals with Google Cloud
- Natural Language Processing on Google Cloud
- Recommendation Systems on Google Cloud

A Professional Machine Learning Engineer builds, evaluates, productionizes, and optimizes ML models by using Google Cloud technologies and knowledge of proven models and techniques. The ML Engineer:

- handles large, complex datasets and creates repeatable, reusable code.
- considers responsible AI and fairness throughout the ML model development process, and collaborates closely with other job roles to ensure long-term success of ML-based applications.
- has strong programming skills and experience with data platforms and distributed data processing tools.
- is proficient in the areas of model architecture, data and ML pipeline creation, and metrics interpretation.
- is familiar with foundational concepts of MLOps, application development, infrastructure management, data engineering, and data governance.
- makes ML accessible and enables teams across the organization.

By training, retraining, deploying, scheduling, monitoring, and improving models, the ML Engineer designs and creates scalable, performant solutions.

Recommended candidate:

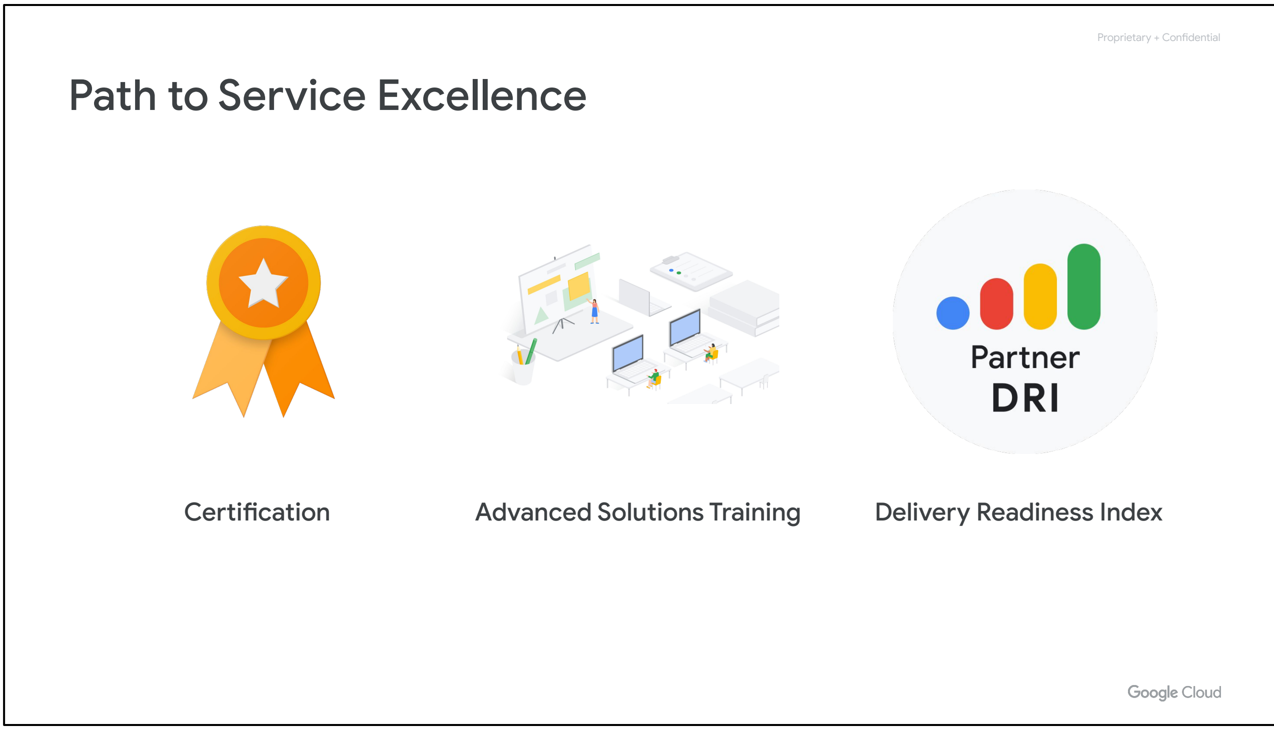
- Has in-depth experience setting up cloud environments for an organization
- Has experience deploying services and solutions based on business requirements

Learner Commitment

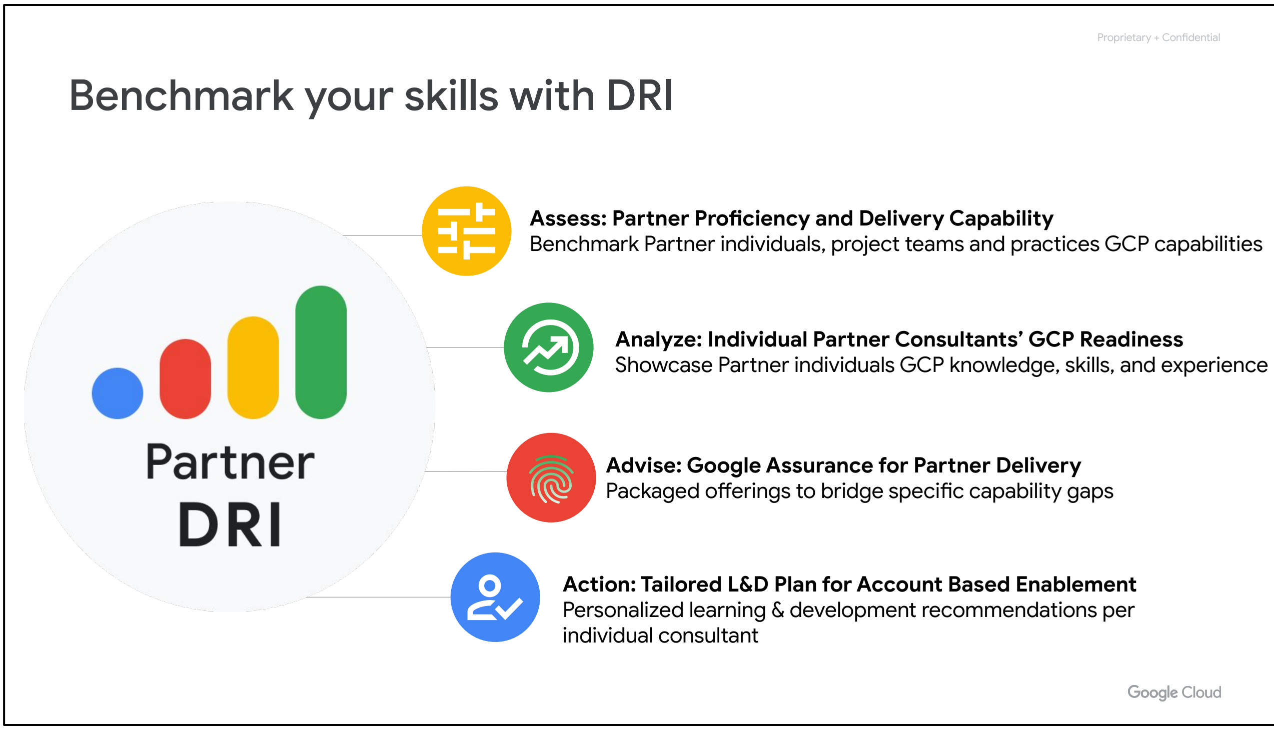
Each week, learners are to complete the learning path's course content, Cloud Skills Boost for Partner Quests/Challenge Labs and material that the mentor has recommended that will support learning.

- **Workshop Day:** Meet for the cohort's weekly 'general session'. **(≅ 2 hours)**
- **During the week:** Complete the week's course, perform hands-on labs, review any additional material suggested material for the week. **(≅ 8 - 16 hours)**

➔ **Important:** Learners must allocate time between each weekly session to study and familiarize themselves with any prerequisite knowledge they may lack. It is also recommended that learners complete the next week's course prior to the scheduled workshop.



Certification is just one step on your professional journey. Google Cloud also offers our partners access to advanced solutions training, and a new quality-focused program called Delivery Readiness Index (DRI) to help you achieve service excellence with your customers.



DRI helps to benchmark partner proficiency and capability at any point during the customer journey however should be used primarily as a lead measure to predict and prepare for partner delivery success.

DRI assesses and analyzes Partner Consultant GCP proficiency by creating a DRI Profile inclusive of their GCP knowledge, skills, and experience.

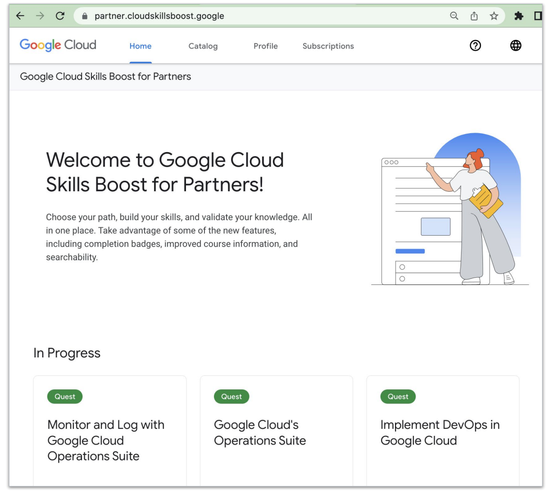
With the DRI insights, we can prescriptively advise the partner project team on the ground and bridge niche capability gaps.

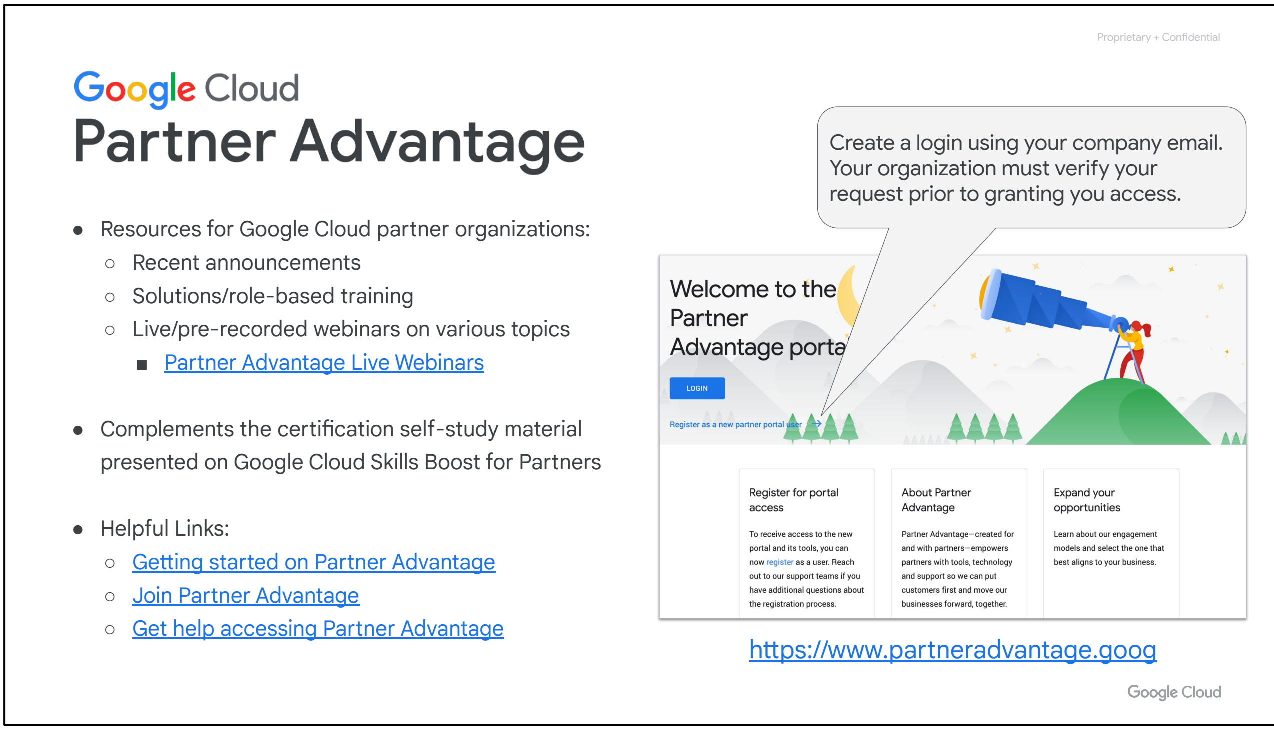
DRI also takes action. For partner consultants, DRI generates a tailored L&D plan that prescribes personalized learning, training, and skill development to build GCP proficiency.

Google Cloud Skills Boost for Partners

https://partner.cloudskillsboost.google/

- On-demand course content
- Hands-on labs
- Skill Badges
- **FREE** to Google Cloud Partners!





The getting started link:
https://support.google.com/googlecloud/topic/9198654#zippy=

Note the top section, **"Getting Started & User Guides"** and two key documents →
Direct Partners to this if they need to enroll into Partner Advantage

1. Logging in to the Partner Advantage Portal - Quick Reference Guide
2. Enrolling in the Partner Advantage Program - Quick Reference Guide

Focus from this point on:

Some context on enrolling in PA:
Access to Partner Portal is given in 2 ways

- Partner Admin Led: Partner Administrator at Partner Company can set up users
- User Led: User can go through Self Registration
  - https://www.partneradvantage.goog/GCPPRM/s/partneradvantageportallogin?language=en_US
  - Or directly to the User Registration Form, https://www.partneradvantage.goog/GCPPRM/s/partnerselfregistration?language=en_US

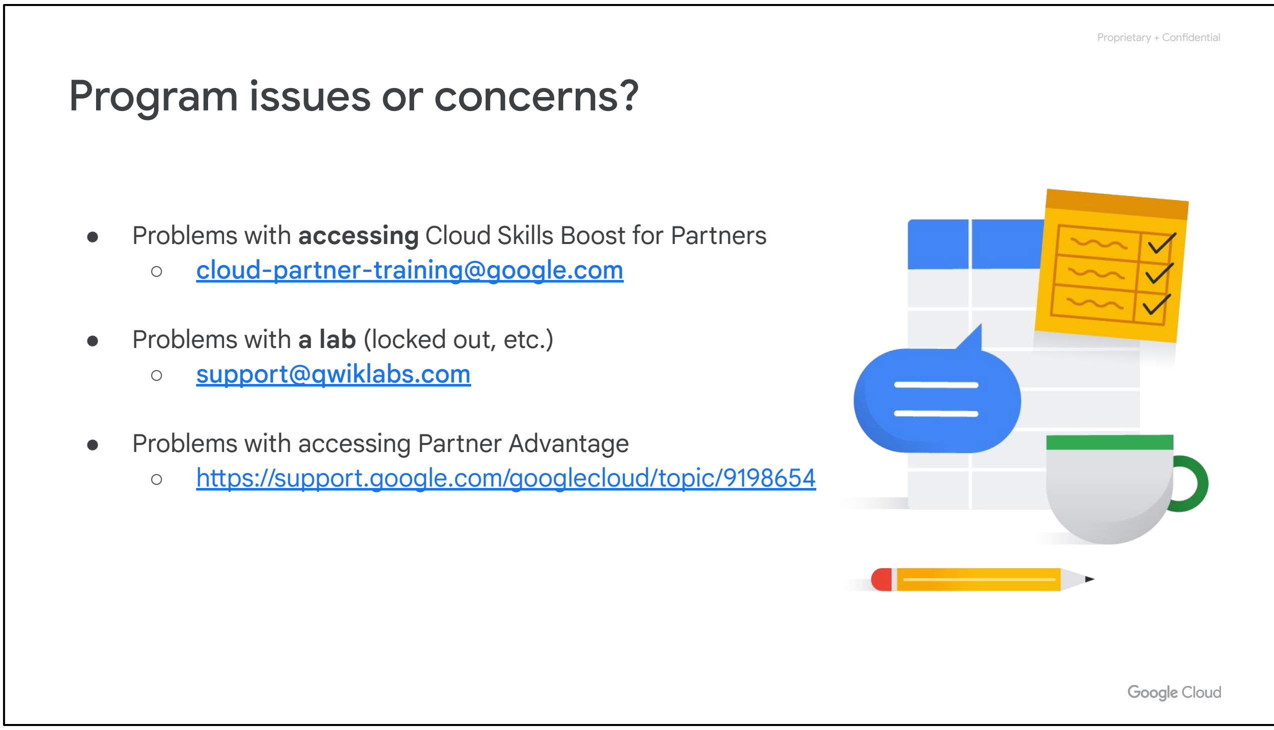
Please Note

- After a user self-registers, they receive an email that essentially states:

    - “Hi {Partner Name}, you are one step away from joining the Google Cloud Partner Advantage Community. Please click to continue with the user registration process. See you in the cloud, The Partner Advantage Team
- Once registered, they can access limited content until their **Partner Administrator approves the user**
- Their Partner Administrator also receive an email notifying them that a member of their organization has registered themselves on their organization’s Google Cloud Partner Advantage account.
    - It also states that this user has limited access to the portal
    - They are provided instructions on how to review and provision the appropriate access for the user that has registered
- Once their admin approves the user, they receive an email that states:
    - Hi {User Name}, Your Partner Administrator has updated your access to the Google Cloud Partner Advantage portal. You have been granted edit access to additional account information on the portal on behalf of your organization to help build your business. For additional access needs, please work with your Partner Administrator. See you in the cloud, The Partner Advantage Team

The net takeaway is, on the Support Page (the first link on this slide) Google Cloud Partner Advantage Support, there’s a section **“Issue accessing Partner Advantage Portal? Click here for troubleshooting steps”**

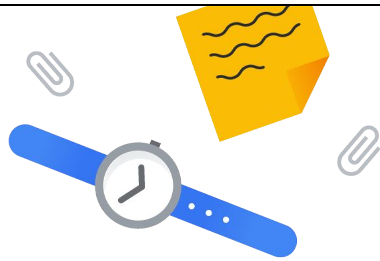
- The source of their issue can be related to the different items shown
- Additionally, there’s a Partner Administrator / Partner Adminstrator Team at their partner organization that has to approve their access.. Until that step is completed, they will have access issues/limitation. They will need to identify who this person or team is at their organization



- Problems with accessing **Cloud Skills Boost for Partners**
  - cloud-partner-training@google.com
- Problems with a **lab** (locked out, etc.)
  - support@qwiklab.com
- Problems with accessing **Partner Advantage**
  - https://support.google.com/googlecloud/topic/9198654

Module 8

Additional Material & Exam Prep



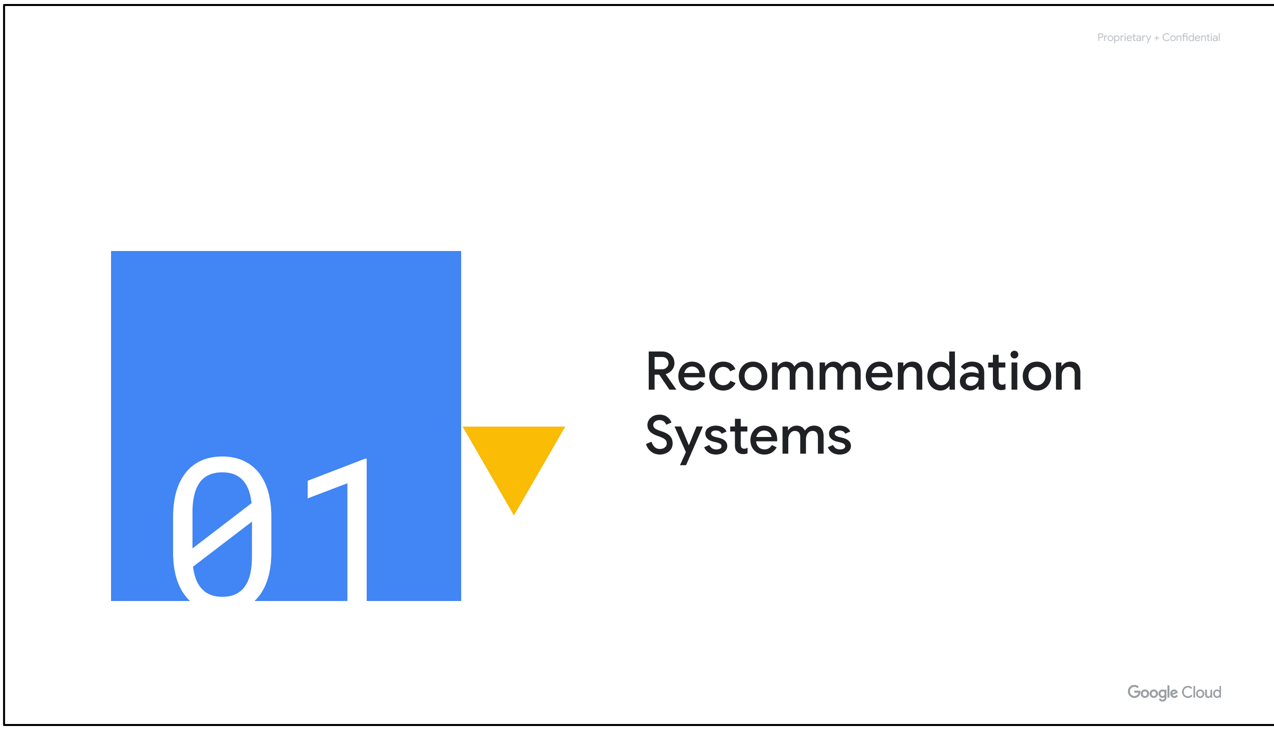
Module Agenda

01 Recommendation Systems

02 Foundation Models: A Quick introduction

03 ML Best Practices & Pitfalls

04 Exam Prep



COURSE:
IMPORTANT: Follow the course "Recommendation Systems with TensorFlow on Google Cloud"

LAB:
"Using neural Networks for Content based Filtering":
https://partner.cloudskillsboost.google/course_sessions/2676369/labs/325069

Proprietary + Confidential

Some real-world recommendation systems

Google Cloud

Source: Custom Screenshots from course [Recommendation Systems with TensorFlow on Google Cloud]()

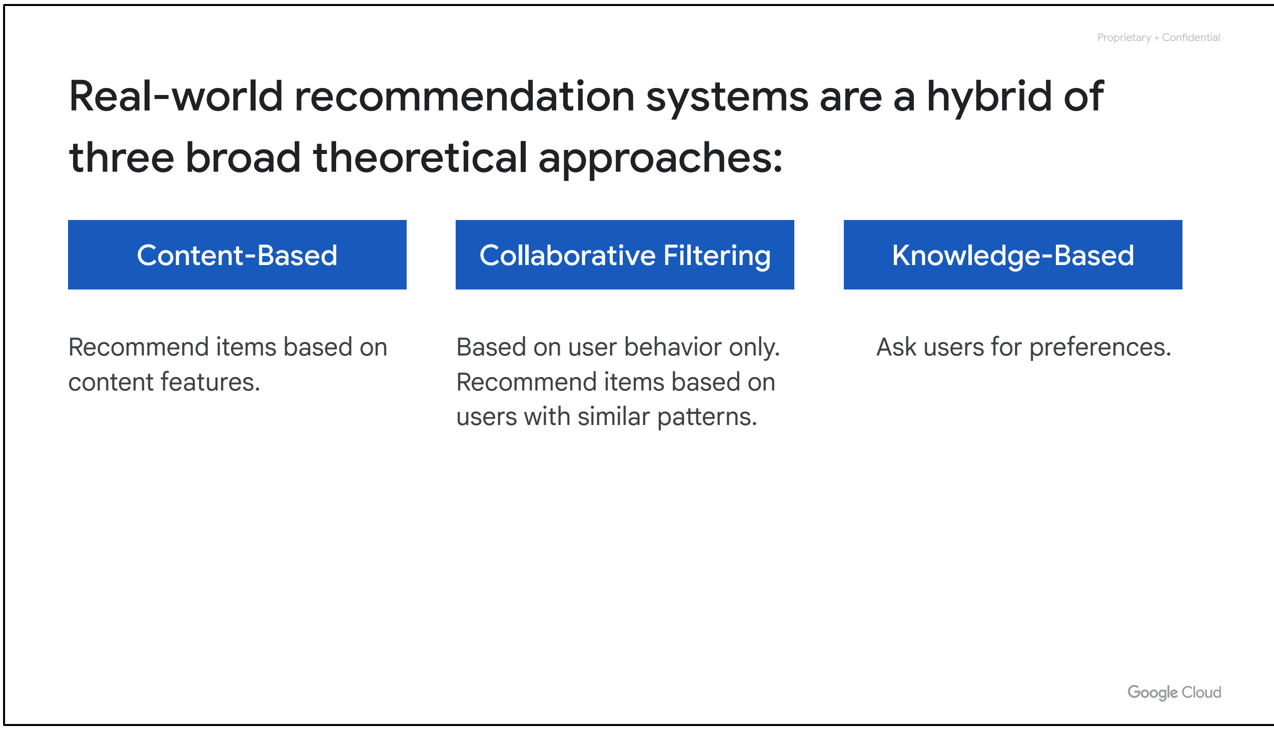
Amazon:
Amazon primarily uses a hybrid recommendation system, which combines collaborative filtering and content-based techniques. Their system is known for its item-to-item collaborative filtering approach. When a user views a product, Amazon recommends other products that are frequently bought together, viewed together, or share similar attributes. Additionally, Amazon incorporates user behavior data, such as browsing history, purchase history, and items in the user's wish list, to provide more personalized recommendations.

Spotify:
Spotify uses a combination of collaborative filtering, content-based, and context-aware techniques to recommend music to its users. Collaborative filtering is used to identify users with similar music tastes and recommend songs, albums, or artists based on their preferences. Content-based methods are employed to analyze features of the music, such as tempo, genre, or mood, and recommend songs with similar characteristics. Finally, context-aware techniques consider factors such as the time of day, user's activity (e.g., working out, commuting), or even the user's mood to provide more relevant recommendations in the form of context-specific playlists like "Morning Commute" or "Workout Mix."

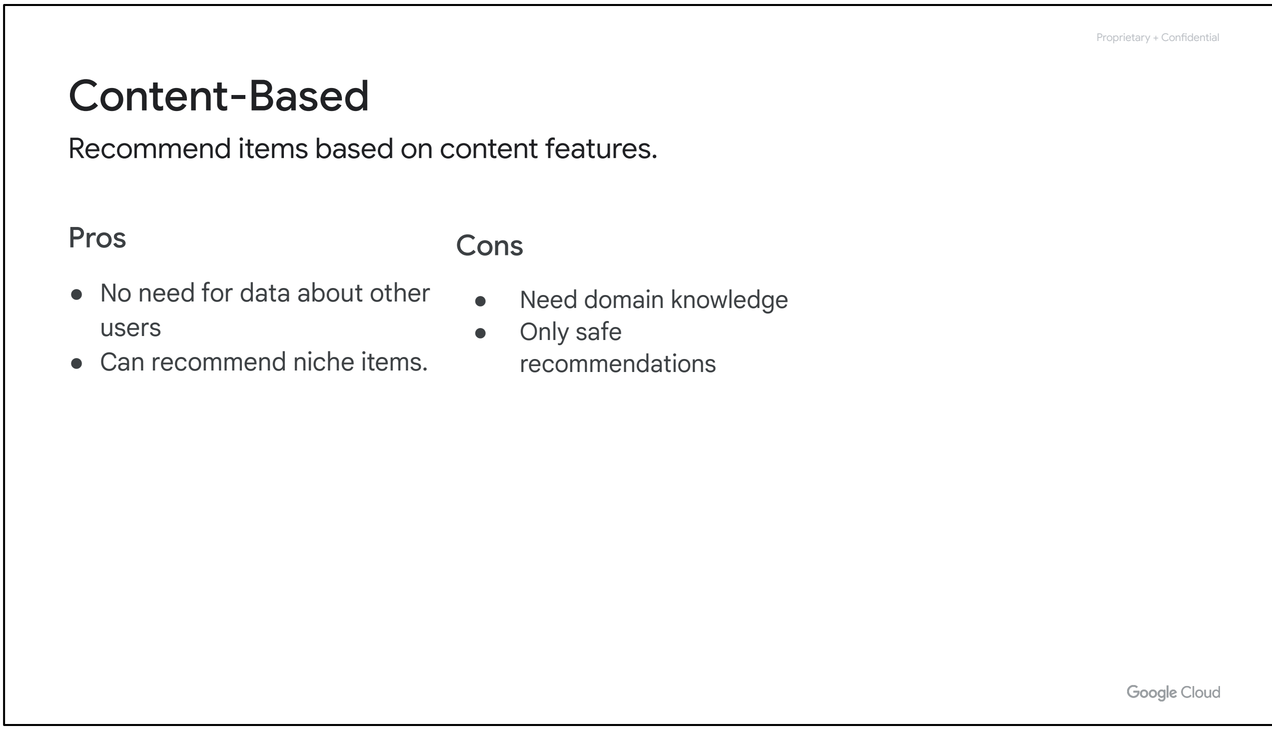
Netflix:

Netflix uses a hybrid recommendation system that combines content-based and collaborative filtering methods to suggest movies and TV shows to its users. Collaborative filtering helps identify users with similar viewing habits and recommends content based on their preferences. Content-based techniques analyze various features of the movies or TV shows, such as genre, actors, directors, or themes, and suggest content with similar attributes. Netflix also incorporates context-aware techniques by considering factors like the time of day, device type, and user's viewing history to provide more personalized and relevant recommendations.

In summary, these popular platforms use a combination of recommendation techniques to provide a personalized experience to their users. By leveraging collaborative filtering, content-based, context-aware, and hybrid approaches, Amazon, Spotify, and Netflix can offer highly accurate and relevant suggestions that cater to individual user preferences and behavior.



Source: Custom Screenshots from course [Recommendation Systems with TensorFlow on Google Cloud]



Content-based recommendation systems:
These recommend items based on the similarity between the content of the items and the user's preferences. They use features such as item descriptions, tags, genres, or other relevant attributes to create user profiles and generate recommendations based on content similarities.

Proprietary + Confidential

Collaborative Filtering

Based on user behavior only. Recommend items based on users with similar patterns.

Pros

- No domain knowledge
- Serendipity
- Great starting point

Cons

- Cold start with fresh items/users
- Sparsity
- No context features

Google Cloud

Collaborative filtering recommendation systems:
Collaborative filtering systems generate recommendations based on the relationships between users and their preferences. There are two main types of collaborative filtering:

> **a. User-based collaborative filtering:** Identifies users who are similar to the target user based on their past behavior or preferences and recommends items that these similar users have liked or interacted with.
> **b. Item-based collaborative filtering:** Identifies items that are similar to each other based on the preferences or interactions of users who have engaged with them and recommends items that are similar to those the target user has previously liked or interacted with.

Knowledge-Based

Ask users for preferences.

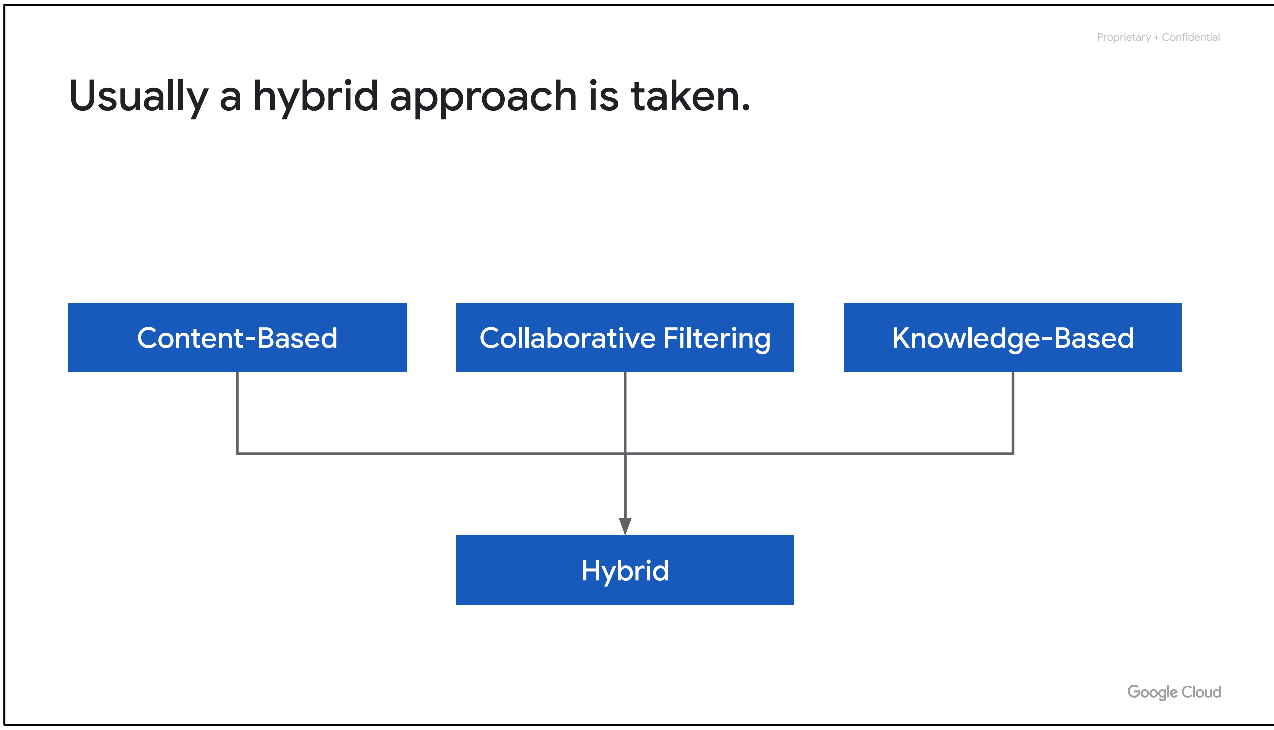
Pros

- No interaction data needed
- Usually high-fidelity data from user self-reporting

Cons

- Need user data
- Need to be careful with privacy concerns

knowledge-based recommendation system:
is a type of recommendation system that leverages domain knowledge or expert knowledge to generate recommendations. This approach is particularly useful when there is limited data about users' preferences or when the available data is sparse, making it challenging to use content-based or collaborative filtering techniques effectively.



Source: Custom Screenshots from course Recommendation Systems with TensorFlow on Google Cloud

02

Foundation Models: A Quick introduction

In August 2021, following the advances by several key papers, Stanford HAI rang a bell to signal what they believed to be a new subfield called foundation models

Example key papers

BERT [Oct 2018]: Pre-text task with ~340M transformer model

XLNet [June 2019]: Autoregressive Pretraining

GPT-3 [May 2020]: Chatbot model at extreme scales (175B)

CLIP [Jan 2021]: Image captioning using pre-training tricks inspired by BERT (63M)

DALL·E [Jan 2021]: Text-to-image generation with a "mini" GPT-3 (12B)

arXiv:2108.07258v3 [cs.LG] 12 Jul 2022

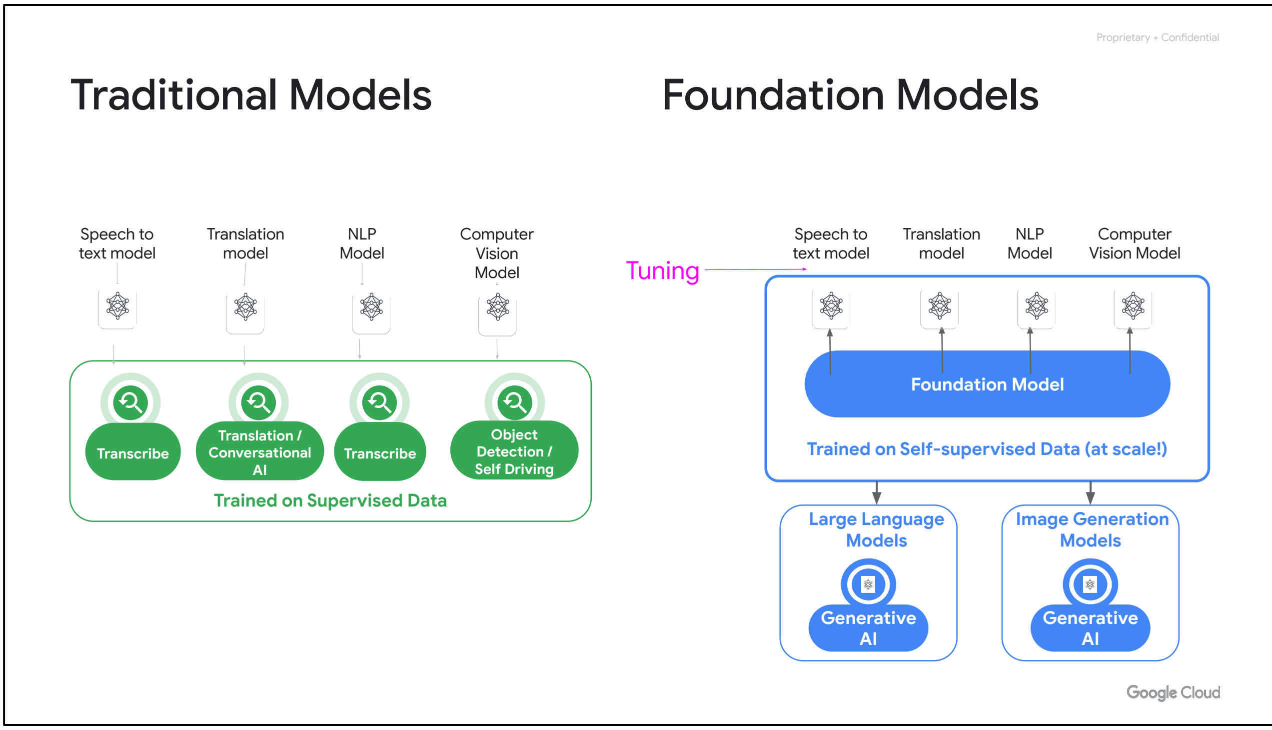
On the Opportunities and Risks of Foundation Models

Rishi Bommasani* Drew A. Hudson Ehsan Adeli Russ Altman Simran Arora Sydney von Arx Michael S. Bernstein Jeannette Bohg Antoine Bosselut Emma Brunskill Erik Brynjolfsson Shyamal Buch Dallas Card Rodrigo Castellon Niladri Chatterji Annie Chen Kathleen Creel Jared Quincy Davis Dorottya Demszky Chris Donahue Moussa Doumbouya Esin Durmus Stefano Ermon John Etchemendy Kawin Ethayarajh Li Fei-Fei Chelsea Finn Trevor Gale Lauren Gillespie Karan Goel Noah Goodman Shelby Grossman Neel Guha Tatsunori Hashimoto Peter Henderson John Hewitt Daniel E. Ho Jenny Hong Kyle Hsu Jing Huang Thomas Icard Saahil Jain Dan Jurafsky Pratyusha Kalluri Siddharth Karamcheti Geoff Keeling Fereshte Khani Omar Khattab Pang Wei Koh Mark Krass Ranjay Krishna Rohith Kuditipudi Ananya Kumar Faisal Ladhak Mina Lee Tony Lee Jure Leskovec Isabelle Levent Xiang Lisa Li Xuechen Li Tengyu Ma Ali Malik Christopher D. Manning Suvir Mirchandani Eric Mitchell Zanele Munyikwa Suraj Nair Avanika Narayan Deepak Narayanan Ben Newman Allen Nie Juan Carlos Niebles Hamed Nilforoshan Julian Nyarko Giray Ogut Laurel Orr Isabel Papadimitriou Joon Sung Park Chris Piech Eva Portelance Christopher Potts Aditi Raghunathan Rob Reich Hongyu Ren Frieda Rong Yusuf Roohani Camilo Ruiz Jack Ryan Christopher Ré Dorsa Sadigh Shiori Sagawa Keshav Santhanam Andy Shih Krishnan Srinivasan Alex Tamkin Rohan Taori Armin W. Thomas Florian Tramèr Rose E. Wang William Wang Bohan Wu Jiajun Wu Yuhuai Wu Sang Michael Xie Michihiro Yasunaga Jiaxuan You Matei Zaharia Michael Zhang Tianyi Zhang Xikun Zhang Yuhui Zhang Lucia Zheng Kaitlyn Zhou Percy Liang*[1]

Center for Research on Foundation Models (CRFM)
Stanford Institute for Human-Centered Artificial Intelligence (HAI)
Stanford University

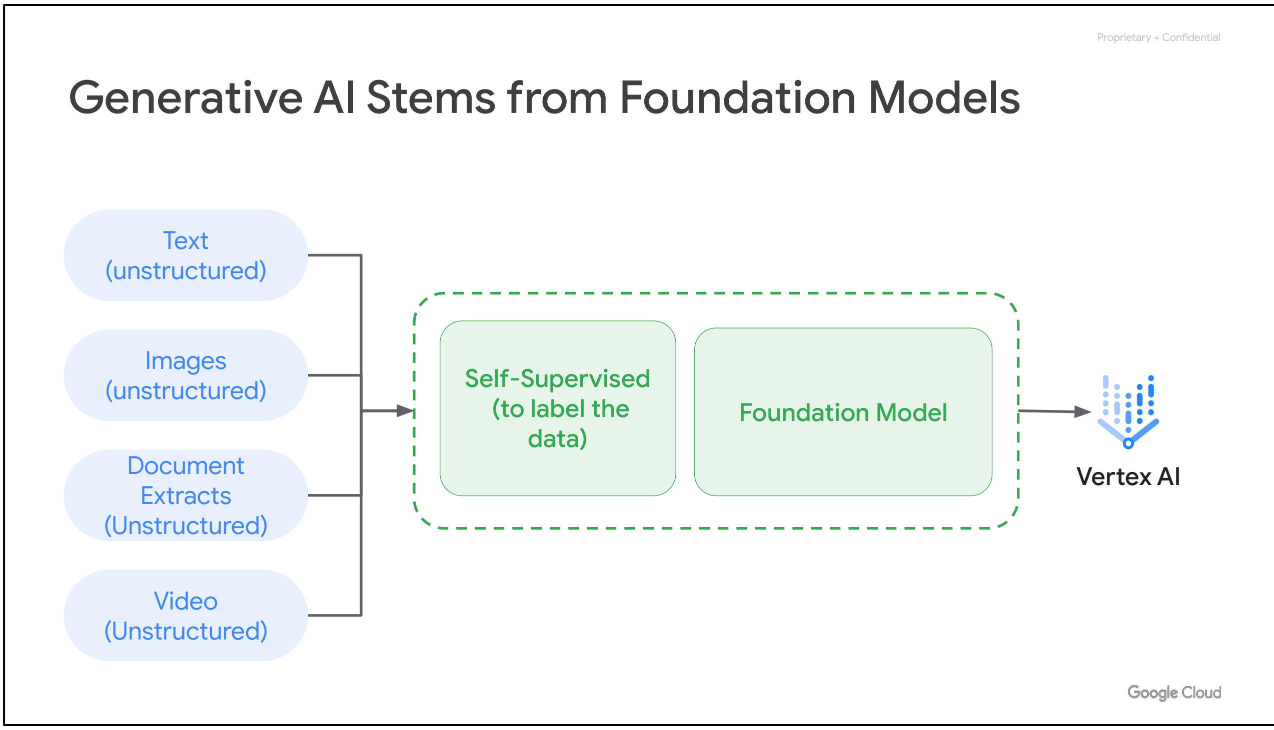
AI is undergoing a paradigm shift with the rise of models (e.g., BERT, DALL-E, GPT-3) trained on broad data (generally using self-supervision at scale) that can be adapted to a wide range of downstream tasks. We call these models foundation models *to underscore their critically central yet incomplete character. This report provides a thorough account of the opportunities and risks of foundation models, ranging from their capabilities (e.g., language, vision, robotic manipulation, reasoning, human interaction) and technical principles (e.g., model architectures, training procedures, data, systems, security, evaluation, theory) to their applications (e.g., law, healthcare, education) and societal impact (e.g., inequity, misuse, economic and environmental impact, legal and ethical considerations). Though foundation models are based on standard deep learning and transfer learning, their scale results in new emergent capabilities, and their effectiveness across so many tasks incentivizes homogenization. Homogenization provides powerful leverage but demands caution, as the defects of the foundation model are inherited by all the adapted models downstream. Despite the impending widespread deployment of foundation models, we currently lack a clear understanding of how they work, when they fail, and what they are even capable of due to their emergent properties. To tackle these questions, we believe much of the critical*

From this deck in Partner Advantage Portal:
https://docs.google.com/presentation/d/1jkK6qgzvV1TcmYaXZY95UVbu7yVgjIO7DdIWpMhjO5A/edit?resourcekey=0-kI4IvihgE-u9hCREQmSRqA#slide=id.g22b2308c44d_8_1302



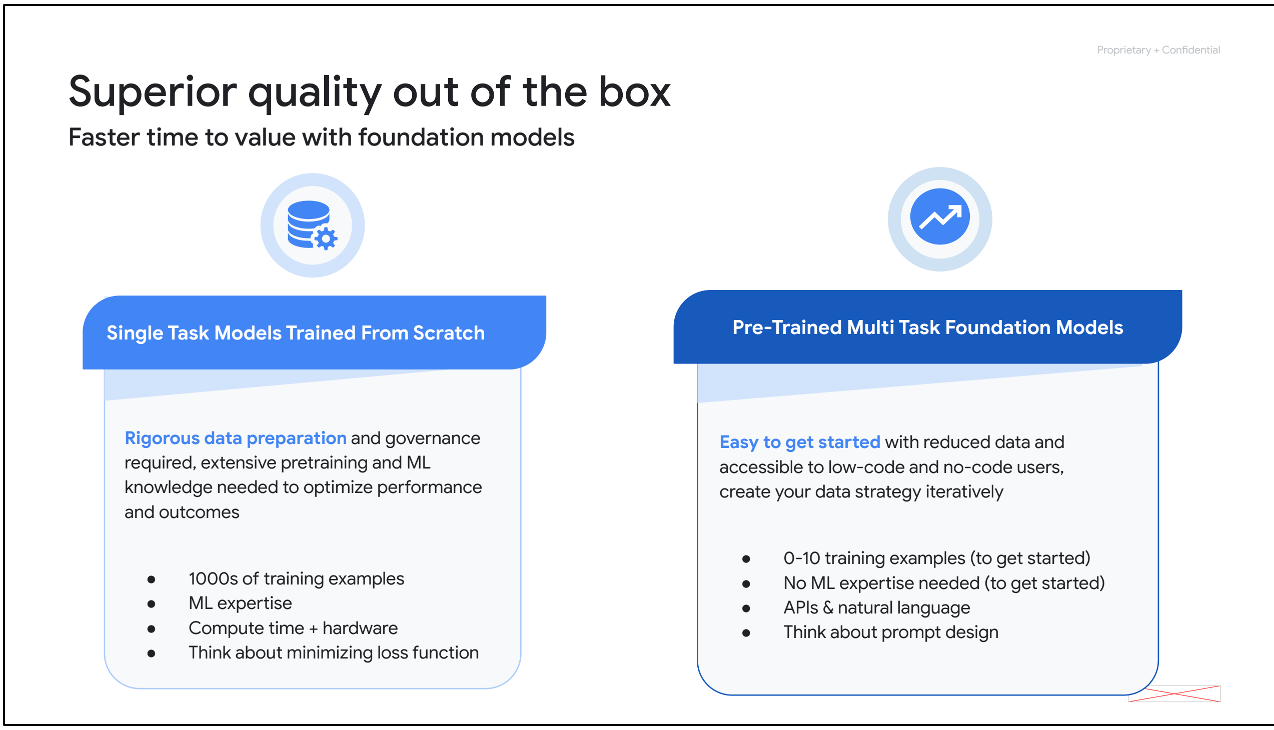
What is a Foundation Model? {Animated Slide)

A foundation model is a large AI model pre-trained on a vast quantity of data that was "designed to be adapted" (or fine-tuned) to a wide range of downstream tasks, such as sentiment analysis, image captioning, and object recognition.



Illustration

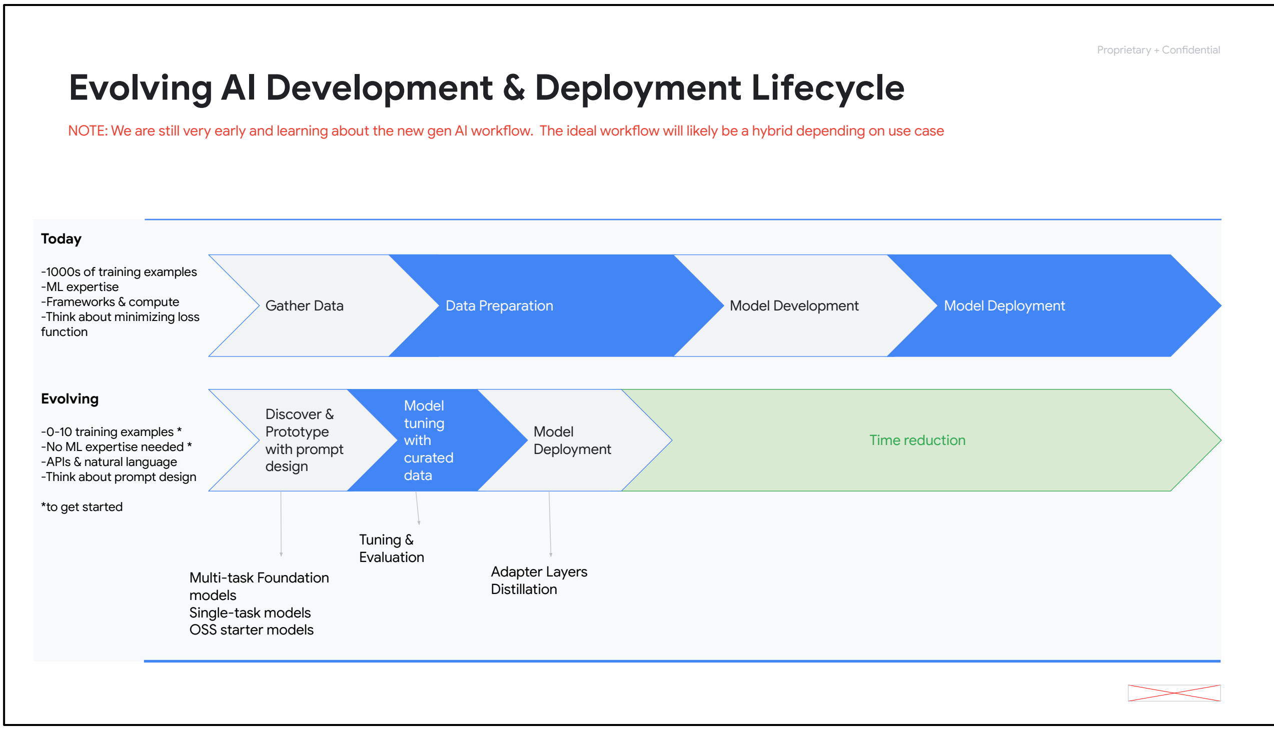
A foundation model is a large AI model pre-trained on a vast quantity of data that was "designed to be adapted" (or fine-tuned) to a wide range of downstream tasks



We expect strong performance OOTB with LLMs, meaning that the old paradigm of having to train with your own data, and consequently having your data strategy in order as a prerequisite for AI is being challenged (for the good)

Superior quality with no uptraining required, Foundation Models can deliver exceptional results with minimal customer data and expertise

Source: https://docs.google.com/presentation/d/1OocpNlkVgZ4l16wz5pl4h_0j8MtK4alpO0_xFMgQm3s/edit?resourcekey=0-Y0UC9U_5N7svKKjgKpGX_g#slide=id.g2595aa9e778_0_5676



Source:
https://docs.google.com/presentation/d/1OocpNlkVgZ4l16wz5pl4h_0j8MtK4alpO0_xFMgQm3s/edit?resourcekey=0-Y0UC9U_5N7svKKjgKpGX_g#slide=id.g2595aa9e778_0_5676

03

ML Best Practices & Pitfalls

Proprietary + Confidential

Reference

Guidelines for developing high quality ML solutions

Cloud Architecture Center

Was this helpful?

Guidelines for developing high-quality ML solutions

Send feedback

Last reviewed 2022-02-17 UTC

This document collates some guidelines to help you assess, ensure, and control quality in machine learning (ML) solutions. It provides suggestions for every step of the process, from developing your ML models to deploying your training systems and serving systems to production. The document extends the information that's discussed in Practitioners Guide to MLOps by highlighting and distilling the quality aspects in each process of the MLOps lifecycle.

This document is intended for anyone who is involved in building, deploying, and operating ML solutions. The document assumes that you're familiar with MLOps in general. It does not assume that you have knowledge of any specific ML platform.

Overview of machine learning solution quality

In software engineering, many standards, processes, tools, and practices have been developed to ensure software

Docs: Guidelines for developing high quality ML solutions

Proprietary + Confidential

Reference

Rules of Machine Learning

Rule #1: Don't be afraid to launch a product without machine learning.

Machine learning is cool, but it requires data. Theoretically, you can take data from a different problem and then tweak the model for a new product, but this will likely underperform basic **heuristics**. If you think that machine learning will give you a 100% boost, then a heuristic will get you 50% of the way there.

For instance, if you are ranking apps in an app marketplace, you could use the install rate or number of installs as heuristics. If you are detecting spam, filter out publishers that have sent spam before. Don't be afraid to use human editing either. If you need to rank contacts, rank the most recently used highest (or even rank alphabetically). If machine learning is not absolutely required for your product, don't use it until you have data.

Rule #2: First, design and implement metrics.

Before formalizing what your machine learning system will do, track as much as possible in your current system. Do this for the following reasons:

1. It is easier to gain permission from the system's users earlier on.
2. If you think that something might be a concern in the future, it is better to get historical data now.
3. If you design your system with metric instrumentation in mind, things will go better for you in the future. Specifically, you don't want to find yourself grepping for strings in logs to instrument your metrics!
4. You will notice what things change and what stays the same. For instance, suppose you want to directly optimize

Docs: [Rules of ML]

Proprietary + Confidential

Avoid these top 10 ML pitfalls

Defining KPI's | Collecting data | Integration | Infrastructure | Optimizing ML

1. ML requires just as much software infrastructure
2. No data collected yet
3. Assume the data is ready for use
4. Keep humans in the loop
5. Product launch focused on the ML algorithm
6. ML optimizing for the wrong thing
7. Is your ML improving things in the real world
8. Using a pre-trained ML algorithm vs building your own
9. ML algorithms are trained more than once
10. Trying to design your own perception or NLP algorithm

What are some of these common pitfalls? I'm glad you asked. So here is our kind of click baity fun, top ten pitfalls organizations hit when they first try ML. And here's a list, very informally I've aggregated after several years of talking with new ML practitioners that come to us and they say, "We're so excited to this great new thing, it's going to be awesome." And then they might fall into some common pitfalls. I've seen it at Google, and I've seen it with our partners as well. First one, perhaps one of the most common, you thought training your own ML algorithm would be faster than writing the software. Usually, this is not the case. And the reason is that to make a great ML system beyond just the algorithm, you're going to need lots of things around the algorithm like a whole software stack to serve, to make sure that it's robust and it's scalable and has great up-time. And all of this, you're going to have to do for software anyway. But then if you try to use an algorithm, you put in additional complexities around data collection, training, all of that just gets little bit more complicated. So usually, we really push people to start with something simpler in software only.

Next one, one of my favorites. You want to do ML, but you haven't collected the data yet. Full stop, you need the data. There's really no use talking about doing great ML if you have not collected great data or you do not have access to great data.

And let's say you do have that data, you've been logging in for years, so it's written on some system that someone in another department controls, but you haven't looked at it, willing to bet that if you haven't looked, that data is not really ready to use, and it goes even beyond that. If there's not someone in your organization who's regularly reviewing that data or generating reports or new insights, if that data is not generating value already, likely, it's not the effort to maintain it is not being put in and data has this kind of magical way of going stale. Of all the clients I've ever talked to, I've never met one who overestimated the amount of effort it would take to collecting clean data. No one has ever said that was easier than I expected, expect there to be a lot of pain and friction here.

What's the next one? You forgot to put and keep humans in the loop. So when we get into these ML systems that start to perform core tasks or core business processes in our organizations, they become really important. And appropriately, organizations become risk averse around these systems because they are the breadwinners of the organization and then becomes very important to mitigate this risk. And one of the myriad of ways we do that is we keep humans inside the loop so that they are reviewing the data, handling cases the ML did not handle very well and curating its training inputs. And we're going to talk about this more later, but this is a feature of every production ML system I know in Google, is that it has humans in the loop.

What about this one? You launched a product whose initial value prop was its ML algorithm instead of some other feature. So this is a problem because A, your users probably don't care if what you're giving them is the ML, they just care if it's got that new cool feature or if its recommendations are really good. And, if you launch something whose initial value prop is just ML, it has new data to operate on. It needs lots of users to generate that data so it may learn how to interact better.

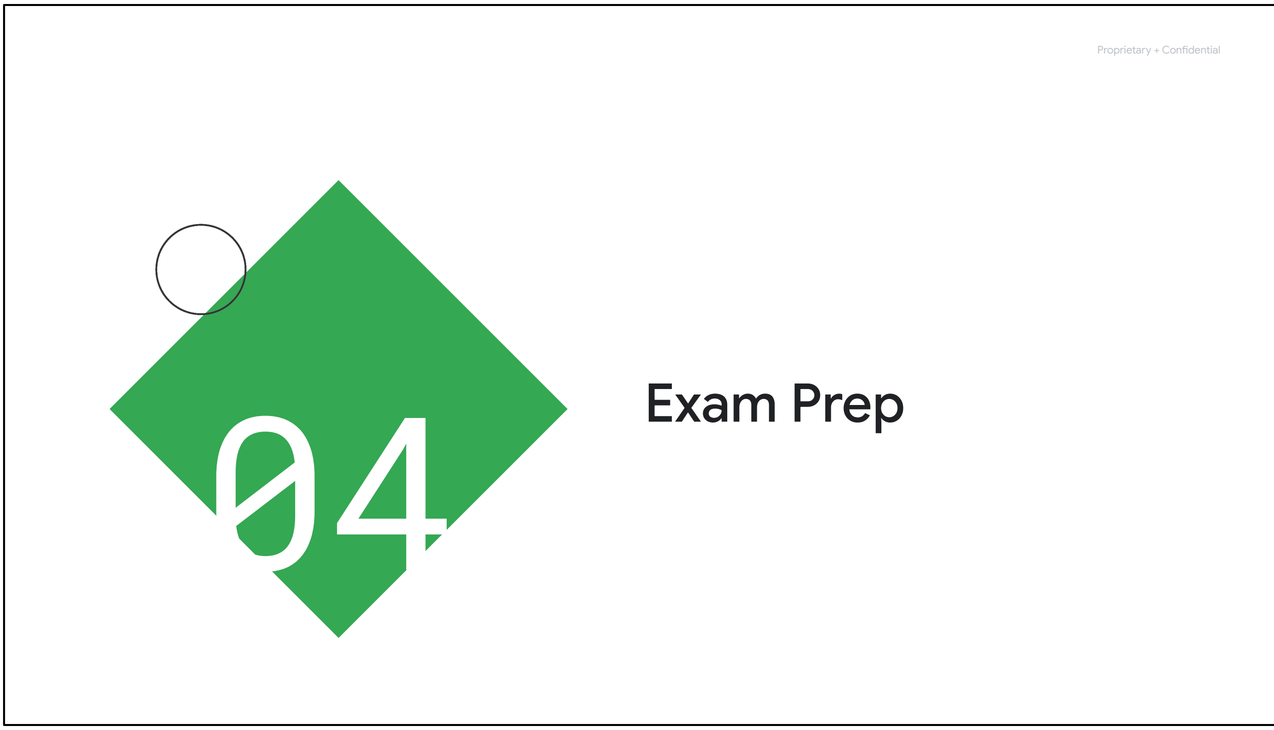
What about you made a great end ML system, it just happens to optimize for the wrong thing. So imagine if Google search was optimizing for, let's say a user engagement as measured by how often someone clicked on search results. It sounds good. We want our users to like our product, we want our users to stay engaged. But if we optimize for how often they click, maybe then the ML algorithm will learn to kind of serve bad content because it forces users to come back, keep clicking. So we always want to be careful about optimizing for something that's pretty good, need not be perfect, but we will always want to look out for perverse incentives.

So what happens if you forget to measure if your ML algorithm is actually improving things in the real world? You put it out there, you turned it on, it serves users, but you can't tell how much better it is, you can't tell if there's any uplifting customer engagement, or lifetime value. That's always really worrisome because then how are you going to go back to your boss or your boss's boss and say, "Hey, I want to do this for another product if you cannot show the impact of the success."

And then I've seen a couple of customers to this next ones, you confuse the ease of use and the value add of somebody else's pre-trained ML algorithm with building your own. So Google Cloud has a couple what we call ML APIs. For instance, with vision, you can send it an image and it will perform image classification on some predefined labels. Well that's great, it's super easy to use. You don't have to worry about any infrastructure, or any training data, or any data collection, very easy to use. It is a very different ballgame than if you went to start to build your own, especially if you want to do your own ML algorithm that does not kind of come pre canned, it's a lot more effort.

We thought after we research that production ML algorithms were trained only once. You're like, "Hey, it's on my laptop, it's doing great on that data set. I'm basically done." No, you're probably about 10 percent of the way through. It turns out that if you're going to have an ML algorithm that's going to be part of your core business processes, it's going to be retrained many, many times and you're going to want to invest the effort to make that process very easy and seamless.

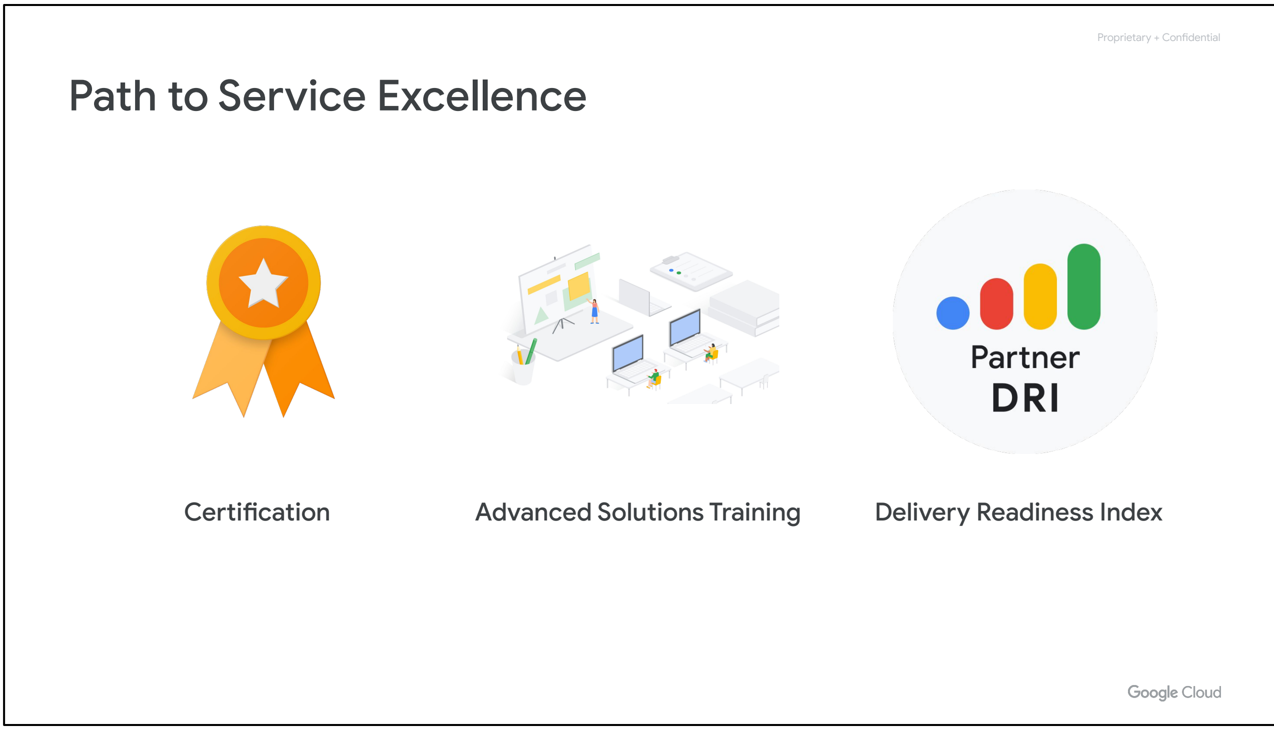
And the final one is actually the only one these I have that addresses a confusion about the challenge involved an opera optimizing the ML algorithm, and that's, you want to design your own in-house perception, i.e. image or speech, or NLP classification, or that's natural language processing. So these are kind of a peculiar pitfall in the sense that they seem they're much easier than they really are. And in fact, all the algorithms we have to address these are very highly tuned from decades of academic research and you should almost always take one off the shelf, already made or already kind of defined, instead of trying to do your own research, it's very expensive.



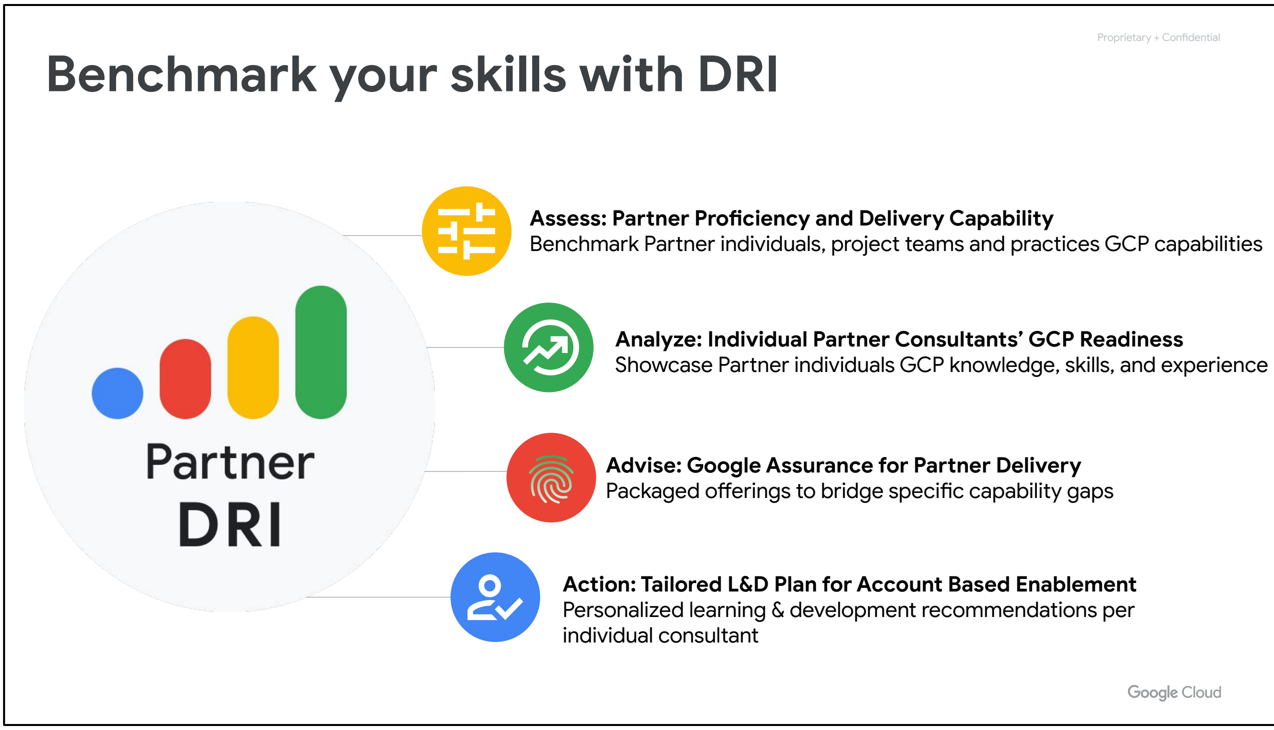
TIP:

- Spend 30 mins or the rest of the time you need
- Use the PMLE Sample Questions:
- Run through about 10 Q&A’s
- **show your approach to these questions and answers (important)**

Same link below
https://docs.google.com/forms/d/e/1FAIpQLSeYmkCANE81qSBqLW0g2X7RoskBX9yGYQu-m1TtsjMvHabGqg/viewform



Certification is just one step on your professional journey. Google Cloud also offers our partners access to advanced solutions training, and a new quality-focused program called Delivery Readiness Index (DRI) to help you achieve service excellence with your customers.



DRI helps to benchmark partner proficiency and capability at any point during the customer journey however should be used primarily as a lead measure to predict and prepare for partner delivery success.

DRI assesses and analyzes Partner Consultant GCP proficiency by creating a DRI Profile inclusive of their GCP knowledge, skills, and experience.

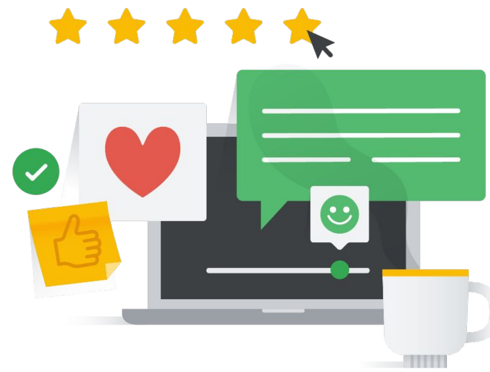
With the DRI insights, we can prescriptively advise the partner project team on the ground and bridge niche capability gaps.

DRI also takes action. For partner consultants, DRI generates a tailored L&D plan that prescribes personalized learning, training, and skill development to build GCP proficiency.

Questions and answers

Thank you for attending this training!

We love your feedback! Please take a minute to complete the survey and help us improve our courses.



Google Cloud