Data Collection + Evaluation

Sourcing and evaluating the data used to train AI involve important considerations. This chapter covers the following questions:

Does our training dataset have the features and breadth to ensure our AI meets our users' needs?

Should we use an existing training dataset or develop our own?

How can we ensure that the data quality is high?

How can we work with labelers to prevent errors and bias in datasets when generating labels?

Are we treating data workers fairly?

Want to drive discussions, speed iteration, and avoid pitfalls? Use the worksheet.

Want to read the chapter offline? Download PDF

What's new when working with AI

In order to make predictions, AI-driven products must teach their underlying machine learning model to recognize patterns and correlations in data. This data is called training data , and can be collections of images, videos, text, audio and more. You can use existing data sources or collect new data expressly to train your system. For example, you might use a database of responsibly crowdsourced data on plants from different regions around the world to train an AI-powered app that recognizes plants that are safe to touch.

The training data you source or collect, and how those data are labeled , directly determines the output of your system — and the quality of the user experience. Once you're sure that using AI is indeed the right path for your product (see User Needs + Defining Success) consider the following:

① **Plan to gather high-quality data from the start**. Data is critical to AI, but more time and resources are often invested in model development than data quality. You'll need to plan ahead as you gather and prepare data, to avoid the effects of poor data choices further downstream in the AI development cycle.

② **Translate user needs into data needs**. Determine the type of data needed to train your model. You'll need to consider predictive power , relevance, fairness, privacy, and security.

③ **Source your data responsibly**. Whether using pre-labeled data or collecting your own, it's critical to evaluate your data and their collection method to ensure they're appropriate for your project.

④ **Prepare and document your data**. Prepare your dataset for AI, and document its contents and the decisions that you made while gathering and processing the data.

⑤ **Design for labelers & labeling**. For supervised learning , having accurate data labels is crucial to getting useful output from your model. Thoughtful design of labeler instructions and UI flows will help yield better quality labels and therefore better output.

⑥ **Tune your model**. Once your model is running, interpret the AI output to ensure it's aligned with product goals and user needs. If it's not, then troubleshoot: explore potential issues with your data.

① Plan to gather high-quality data from the start

Data is critical to AI, but the time and resources invested in model development and performance often outweigh those spent on data quality.

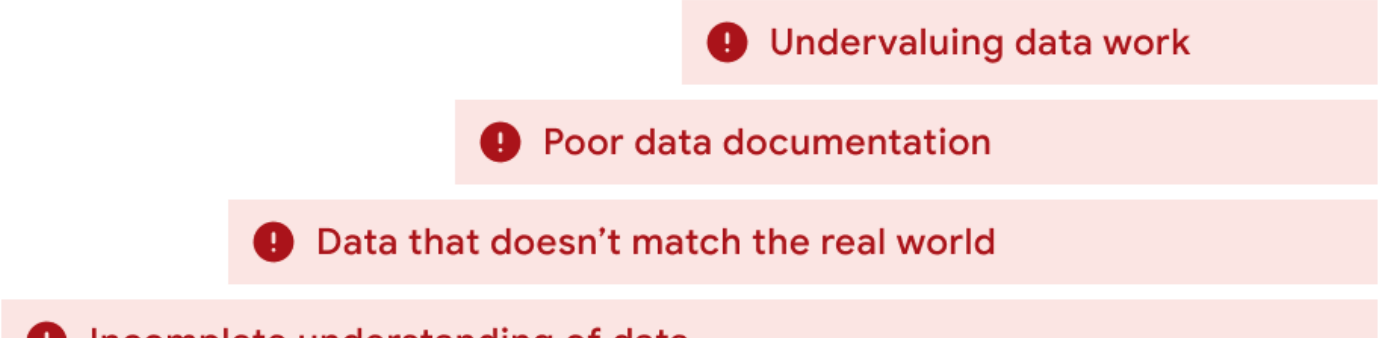
This can lead to significant downstream effects throughout the development pipeline, which we call “ data cascades ”. Here’s a hypothetical example of a data cascade:

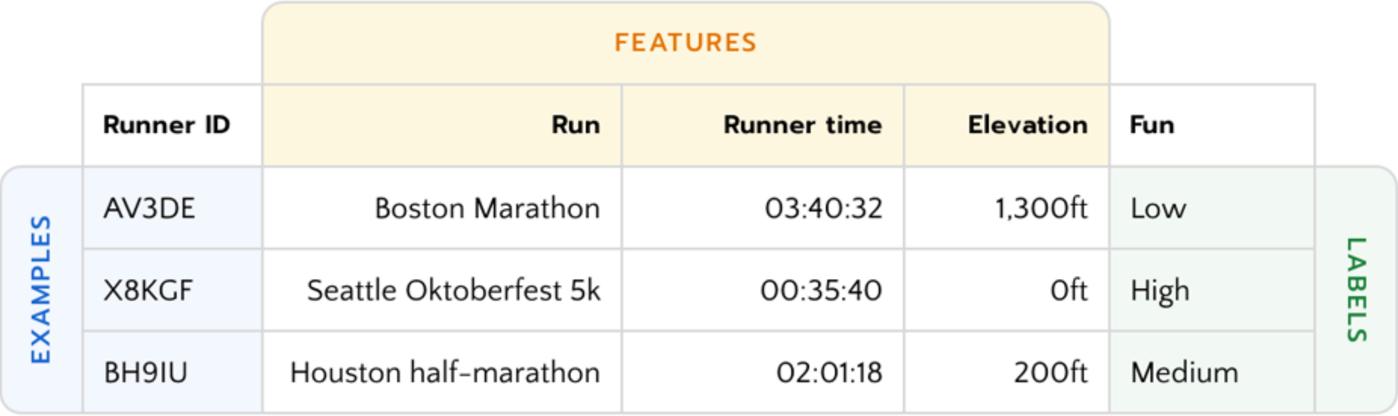
> Let’s say you’re developing an app that allows the user to upload the picture of a plant, and then displays a prediction for the plant’s type, and whether it is safe for humans and pets to touch and eat it.
>
> When you prepared a dataset to train the image classification model, you used mostly images of plants native to North America because you found a dataset that was already labeled and easy to use to train the model.
>
> Once you released the Plant Pal app, however, you found out that many users were reporting plant detection errors in South America.
>
> This is an example of a data cascade , as the effects of the mismatch between training data and user data in the real world were delayed. This cascade could have been avoided by including images of plants native to South America when developing the AI model, or releasing the app to users in North America only.

Some data cascades may be hard to diagnose, and you may not see their impact until users report a poor experience.

It’s best to plan to use high-quality data from the beginning, whether you’re creating a dataset from scratch or reusing existing datasets, and good planning and interrogating your dataset can help you detect issues earlier. High-quality data can be defined as:

- Accurately representing a real-world phenomenon or entity
- Collected, stored, and used responsibly
- Reproducible
- Maintainable over time
- Reusable across relevant applications
- Having empirical and explanatory power


Undervaluing data work
Poor data documentation
Data that doesn’t match the real world
Incomplete understanding of data



		FEATURES				
	Runner ID	Run	Runner time	Elevation	Fun	
EXAMPLES	AV3DE	Boston Marathon	03:40:32	1,300ft	Low	LABELS
	X8KGF	Seattle Oktoberfest 5k	00:35:40	0ft	High	
	BH9IU	Houston half-marathon	02:01:18	200ft	Medium	

The scope of features, the quality of the labels, and representativeness of the examples in your training dataset are all factors that affect the quality of your AI system.

The table above contains data about races that an app could use to train an ML model to predict how enjoyable a given race will be. Here's how examples, features and labels could affect the quality of that model:

Examples

If examples used to train the run recommendation algorithm only come from elite runners, then they would likely not be useful in creating an effective model to make predictions for a wider user base. However, they may be useful in creating a model geared towards elite runners.

Features

If the elevation gain feature was missing from the dataset, then the ML model would treat a 3.0 mile uphill run equally to a 3.0 downhill mile run, even though the human experience of these is vastly different.

Once you’ve defined the data requirements, you’ll start gathering the data.

A good place to start is to determine if there are any existing datasets that you can reuse.

- Do you have access to existing datasets that meet your project requirements? Explore Dataset Search to start looking for available datasets.
- Can you acquire an existing dataset by partnering with another organization, purchasing a dataset, or using client data?

If you decide to acquire an existing dataset, make sure that you have the following information:

- Is this data appropriate for your users and use case?
- How was the data collected?
- Which transformations were applied to it?
- Do you need to augment it with additional data sources to be useful?
- Were any trade-offs and assumptions made when creating it?
- What are the data compliance standards and licensing information for the dataset?
- Does the dataset have any documentation, such as a Data Card?

Models can also make poor predictions due to underfitting, where a model hasn't properly captured the complexity of the relationships among the training dataset features and therefore can't make good predictions with training data or with new data.

There are many resources that can help the software engineers and research scientists on your team with understanding the nuances of training ML models so you can avoid overfitting and underfitting, for example these from Google AI. But first, involve everyone on your product team in a conceptual discussion about the examples, features, and labels that are likely required for a good training set. Then, talk about which features are likely to be most important based on user needs.

2
1
3

2
1
3