Data Collection + Evaluation

Chapter worksheet

Instructions

Block out time to get as many cross-functional leads as possible together in a room to work through these exercises & checklists.

Exercises

1. Get to know your data [~1 hour]

Decide what kind of data you need, whether or not it already exists, and understand the sources.

2. Dataset essential checklist [~30 minutes]

Walk through the checklist to prepare your data for use in your AI product.

3. Design for your raters [~1 hour]

Understand who labels your data and why, and how you can help them do it well.

4. Model tuning with user feedback [~30 minutes]

Get a feel for your users' comfort levels around data usage for your product purposes.

1. Get to know your data

The first task your team has to complete is to identify the type and scope of data needed to train an ML model that can meet your users' needs.

User needs & data needs template

Use this template for each unique user need your ML model will impact.

Example: building a recipe recommendation service that suggests new dishes to cook.

User needs & data needs	
Users	*Home chefs*
User need	*Try new recipes or cuisines*
User action	*Cook a new dish using the recipe based on recommendation*
ML system output	*Recommendations for new recipes*
ML system learning	*Patterns of behavior around choosing recipe recommendations*
Training dataset needed	*Set of recipes user has previously found, used, and liked*
Key **features** needed in dataset	*Ingredient cost* *Cuisine type* *Allergens*

	Dietary restrictions
Key **labels** needed in dataset	*Home cook's accept / reject of recommended recipe* *Home cook's feedback as to why suggestion rejected (user-generated label)* *Recipe ratings from other users*
Data source key user questions	*"How does the app know what I like?"* *"Where do these recipes come from?"* *"How certain is the app in its suggestions?"*

For labels especially, you might need to go deeper with your discussion. For example, let's say you are building the recipe recommendation app above. Cost for ingredients may be a feature you want to include in your ML model. Take a moment to reflect on all the many ways cost can be labeled.

Cost
1257.95
$401,952.1
Two thousand eighty & 2 cents
62 USD
Fourteen shillings and sixpence

Understand user attitudes towards data use

Early in ML model development, set up research sessions to find out what data your users think is necessary and appropriate for the ML model to factor into its decision. The Feedback + control chapter has additional details about collecting data from users.

A common user question you should be ready to answer for any ML feature is:

> *“How does it know?”*

As a first step, it's important to understand what data users *think* your feature uses to make decisions. This is a helpful exercise to prepare for asking users to opt-in to data collection, for explaining your system to users, and for building trust.

For example, you could plan a card sort with users to see what data they would share with your feature versus a trusted, human expert.

Have cards printed or digitally available for participants to sort, ideally as a think-aloud exercise. After they're finished sorting the cards, ask them questions like the ones below:

Research protocol questions

- I see you chose to share these data with the trusted expert, but not the product, tell me about that.
- Are there any data you feel is missing that would improve your experience with the product or trusted expert?
- Of the data you sorted for the feature, is any of the data more or less important for the product to know?
- Take a moment to sort the data by priority and I’ll check back in with you when you’re done.

2. Dataset essential checklist

Once your team knows what data, at a high level, will be required to train your model, you'll need to determine if you can get those data from:

- An existing dataset
- A new dataset
- A combination of an existing and new dataset

Use the checklist below to promote dataset diligence.

Dataset essentials checklist

- ❏ If you need to create a **new dataset,** how are you planning to collect the data?
  - ❏ Answer:
- ❏ If you have an **existing dataset**, what, if any changes or additions need to be made for your user population?
  - ❏ Answer:
- ❏ Use the Facets tool to **evaluate the dataset for bias**.
  - ❏ Date completed:
  - ❏ Insights:
- ❏ How will your dataset **stay up to date** over time?
  - ❏ Answer:
- ❏ Schedule a recurring meeting for your team to discuss tuning your model.
  - ❏ Dates:
- ❏ Schedule time with your product counsel to get legal sign-off on the planned usage of your dataset.
  - ❏ Date of legal review:

- ❏ Agree on what data will be used for your training and test sets.
  - ❏ Training data source (data your model will learn from):
  - ❏ Testing data source (data that your model hasn’t seen before):

3. Design for your raters

If your feature uses supervised learning and you are using a new dataset, you need to understand more about the people who will be teaching or evaluating your model, also known as "raters". Use the template below to get to know your rater population.

Who are your raters?

- ❏ How diverse is your rater pool?
- ❏ Are there particular perspectives or biases that they may be bringing to this task that could impact the quality of the labels?

What is their context and incentive?

- ❏ Are they professionals being paid to do this work or are they volunteers?
- ❏ In what context will they be encountering this task?
- ❏ What is their incentive to complete their rater task?
- ❏ Is there a high risk that they might complete the task incorrectly due to issues like boredom, repetition or poor incentive design?

What tools are they using?

- ❏ Are you designing a rating tool from scratch, or using an existing one? Keep in mind that tools for labeling can range from in-product prompts to specialized software.
- ❏ (In-product) Does the prompt explain the benefit to the user and make it easy for them to provide correct information?
- ❏ (Specialized) Has the software been usability tested?
    - ❏ Are instruction unambiguous?
    - ❏ How quickly and easily can raters read and comprehend instructions?
    - ❏ Can raters complete key tasks quickly and without errors?
    - ❏ Have you provided shortcuts when possible?
    - ❏ Can raters easily correct a mistake?

If your team will be using professional raters, schedule a training meeting to help everyone get on the same page about the user experience of your feature. When you're building tools for professional raters, the article First: Raters offers some useful recommendations.

Rater training UX meeting

Date:

Attendees:

Research with your raters

As part of any training you do with your raters, take time to do research with your raters to validate and iterate on the tools they will be using. Remember to read the article First: Raters to prepare your protocol. Some questions you may want to ask are:

Research protocol questions

- What is clear or confusing about the tool you are using for this task?
- Have you had a task in the past week that had an error? If yes, how did you troubleshoot?
- Can you tell me about a time where you changed your mind after completing a task and went back to change your original rating?

4. Model tuning with user feedback

Once your model has been trained and is working, you'll need to do a lot of testing to help you tune the model. This will involve both technical testing with new test datasets and other tools, as well as getting feedback from users.

Avoiding pitfalls

- ❑ Have we inspected our model with the What-if Tool?
- ❑ What are secondary effects from your reward function that the team might not have planned for? Is there any way to detect and measure them during our pre-launch testing?
  - ❑ Answer:
- ❑ What are the worst case scenarios if something goes wrong with our AI?
  - ❑ Answer:

Pre-launch testing plan

- ❑ Do we have a diverse set of trusted users willing to use our in-development AI and give us feedback?
- ❑ Do we have time and resources set aside for model tuning based on the testing period?
- ❑ Have we agreed on a success metric that will determine launch readiness? (see also User needs + defining success)