

Exercise 1: Start Clearly and Simply

Write what you'd like the machine learned model to do.

We want the machine learned model to:

Mark those email in the Gmail inbox as "important" which are worthy to do so.

Tips for Success

At this point, the statement can be qualitative, but make sure this captures your real goal, not an indirect goal. If you are unsure, take a look at the previous lessons, and see what machine learned models can possibly do.

Exercise 2: Your Ideal Outcome

Adding your ML model to your system should produce a desirable outcome. What is this outcome, independent of the model itself? Note that this outcome may be quite different from how you assess the model and its quality.

Our ideal outcome is:

The ideal outcome is automatically marking/classifying new emails as "important" and "non-important" when they arrive at the inbox. This will redirect the attention of a user to those emails first that are important to him/her.

Tips for Success

You don't need to limit yourself to metrics for which your product has already been optimizing (those will be covered in the next exercise). Instead, try to focus on the larger objective of your product or service.

Exercise 3: Your Success Metrics

Write down your metrics for success and failure with the ML system. The failure metrics are important, that is, how will you know whether the ML system has failed? Both the success and failure metrics should be phrased independently of evaluation metrics for the model. For example, don't talk about precision, recall, or AUC; talk about the anticipated outcomes, instead. Frequently these metrics will be tied to the ideal outcome you specified above.

Our success metrics are:

We can define our success-failure metric as the number of email labels (important/non-important label) that the user has been changed after the marking done by the model.

Our key results (KR) for the success metrics are:

Type your response to exercise 3b here!

Our ML model is deemed a failure if:

If the user altered more than 30% of the labels, that can be counted as the failure of the model.

Tips for Success

- Are the metrics measurable?
 - How will you measure them?
 - It's okay if this is via a live experiment. Many success metrics can't be captured offline. When deciding on your metrics, think about the ideal outcome that you specified in the previous step.
 - When will you be able to measure them?
 - How long will it take for you to know whether your new ML system is a success or failure?
 - Consider engineering and maintenance costs over the long-term.
 - Failure may not only be caused simply by non achievement of a success metric.
-

Exercise 4: Your Output

Write the output that you want your ML model to produce.

The output from our ML model will be:

Predict whether a particular email is important or not.

The output is defined as:

If the probability of the email being important is greater than 0.5 then mark as important.

Tips for Success

- The output must be quantifiable with a definition that machine can produce.
- Are you able to obtain example outputs to use for training data?
 - How and from what source?
- Your output examples may need to be engineered, such as turning watch time into a percentile.
- If it is difficult to obtain example outputs to use for training, you may need to revisit your responses to past exercises to reformulate your problem and goals into ones that will allow you to train a model on your data.

Exercise 5: Using the Output

Write when your output must be obtained from the ML model, and how it is used in your product.

The output from the ML model will be made:

The output from the ML model will be made to auto-tag the important emails.

The outcome will be used for:

Type your response to exercise 5b here!

Tips for Success

Consider how you will use the predicted outcome in your product.

- Will it be presented immediately to the user in a UI?
- Will it be consumed by subsequent business logic?
- What latency requirements do you have?

Those requirements (serving requirement of the ML model) can impact what information can be used to make predictions. For example:

- The latency of using data from remote services may make them infeasible to use.
- If data sources lag in making new information available:
 - Processed logs may be generated only once a day.
 - Certain information is not known until it actually happens (such as conversion events).

The Oracle Test: assume you always had the correct answer. How would you use that in your product?

Exercise 6: Your Heuristics

Write how you would solve the problem if you didn't use ML. For example, what heuristics you might use.

If we didn't use ML, we would:

If we didn't use ML, we would categorize an email as important based on the sender data and reading time data. Briefly explaining, if the user spends a considerable amount of time to read the emails sent from some specific sender then we can mark it as important.

Tips for Success

Think about a scenario where you need to deliver the product tomorrow, and you can only hardcode the business logic. What would you do?

Click the button below to either print or save your responses as a .pdf.

Print Page