Lesson 4 - Goals of Intelligent Systems

Criteria for a Good Goal

1. **Clearly communicate** the desired outcome to all participants. Everyone should be able to understand what success looks like and why it is important, no matter what their background or technical experience.
2. **Be achievable.** The goal can be difficult, but the team should be able to explain the approaches and their feasibility that it would work.
3. **Be measurable.** The evaluation process of an intelligent system is very important to determine if the goal is correct. Some goals will seem perfect to some participants but make no sense to others. Some will align with positive impact but be impossible to measure or achieve.

An Example of Why Choosing Goals Is Hard

Consider an anti-phishing feature backed by an Intelligent System. One form of phishing involves websites that look like legitimate banking sites but are fake sites, controlled by abusers. Users are lured to these phishing sites and tricked into giving their banking passwords to criminals. Not good.

So what should an Intelligent System do? Talk to a machine-learning person and it won't take long to get them excited. They'll quickly see how to build a model that examines web pages and predicts whether they are phishing sites or not. These models will consider things like the text and images on the web pages to make their predictions. If the model thinks a page is a phish, block it. If a page is blocked, a user won't browse to it, won't type their banking password into it. No more problem. Easy. Everyone knows what to do.

So the number of blocks seems like a great thing to measure—block more sites and the system is doing a better job.

Or is it? What if the system is so effective that phishers quit? Every single phisher in the world gives up and finds something better to do with their time?

Perfect But then there wouldn't be any more phishing sites and the number of blocks would drop to zero. The system has achieved total success, but the metric indicates total failure.

Not great. Or what if the system blocks one million phishing sites per day, every day, but the phishers just don't care? Every time the system blocks a site, the phishers simply make another site. The Intelligent System is blocking millions of things, everyone on the team is happy, and everyone feels like they are helping people—but the same number of users are losing their credentials to abusers after the system was built as were losing their credentials before it was built.

Not great. One pitfall with defining success in an Intelligent System is that there are so many things that can be measured and optimized. It's very easy to find something familiar to work with, choose it as an objective, and get distracted from true success.

Recall the three properties of a success criterion:

1. Communicate the desired outcome
2. Be achievable
3. Be measurable

Using the number of blocked phishing pages as a success metric hits #2 and #3 out of the park but fails on #1.

The desired outcome of this system isn't to block phishing sites—it is to stop abusers from getting users' banking passwords.

Types of Goals

1. **Organizational Objectives.** Organizational objectives are the real reason for the Intelligent System. In a business, these might be things like revenue, profit, or the number of units sold. In a nonprofit organization, these might be trees saved, lives improved, or other benefits to society.
2. **Leading Indicators.** Leading indicators are a way to bridge between organizational objectives and the more concrete properties of an Intelligent System (like user outcomes and model properties). If an Intelligent System gets better, customers will probably like it more.

There are two main types of leading indicators: customer sentiment and customer engagement.

Customer sentiment is a measure of how your customers feel about your product. The sentiment is a fuzzy measure because users' feelings can be fickle. It can also be very hard to measure sentiment accurately—users don't always want to tell you exactly what you ask them to tell you. Still, swings in sentiment can be useful indicators of future business outcomes, and Intelligent Systems can certainly affect the sentiment of users who encounter them.

Customer engagement is a measure of how much your customers use your product. This could mean the frequency of usage. It could also mean the depth of usage, as in using all the various features your product has to offer.

3. **User Outcomes.** Another approach for setting goals for Intelligent Systems is to look at the outcomes your users are getting. For example:

- If your system is about helping users find information, are they finding useful information efficiently?
- If your system is about helping users make better decisions, are they making better decisions?
- If your system is about helping users find the content they will enjoy, are they finding content that they end up liking?
- If your system is about optimizing the settings on a computer, are the computers it is optimizing faster?
- And if your system is about helping users avoid scams, are they avoiding scams?

4. **Model Properties**. Within every Intelligent System there are concrete, direct things to optimize, for example:

- The error rate of the model that identifies scams.
- The probability a user will have to re-toast their bread.

- The fraction of times a user will accept the first recommendation of what content to use.
- The click-through rate of the ads the system decides to show.

Ways to Measure Goals

1. **Waiting for more information.** Sometimes it is impossible to tell if an action is right or wrong at the time it happens, but a few hours or days or weeks later it becomes much easier. As time passes, you'll usually have more information to interpret the interaction. Here are some examples of how waiting might help:

- The system recommends content to the user, and the user consumes it completely—by waiting to see if the user consumes the content, you can get some evidence of whether the recommendation was good or bad.
- The system allows a user to type their password into a web page by waiting to see if the user logs in from eastern Europe and tries to get all their friends to install malware, you can get some evidence if the password was stolen or not.

Waiting can be a very cheap and effective way to make a success criterion easier to measure, particularly when the user's behavior implicitly indicates success or failure.

There are a couple of downsides. First, waiting adds latency. This means that waiting might not help with optimizing or making fine-grained measurements. Second, waiting adds uncertainty. There are lots of reasons a user might change their behavior. Waiting gives more time for other factors to affect the measurement

2. **A/B Testing.** Showing different versions of the feature/intelligence to different users can be a very powerful way to quantify the effect of the feature.

3. **Hand Labeling.** You can hire some humans to periodically examine a small number of events/interactions and tell you if they were successful or not. In many cases, this hand labeling is easy to do and doesn't require any skill or training.

To hand-label interactions, the Intelligent System needs to have enough telemetry to capture and replay interactions. This telemetry must contain enough detail so a human can reliably tell what happened and whether the outcome was good or not (while preserving user privacy). This isn't always possible, particularly when it involves having to guess what the user was trying to do, or how they were feeling while they were doing it.

4. **Asking Users.** Perhaps the most direct way to figure out if something is succeeding or not is to ask the user. For example, by building feedback mechanisms right into the product:

- The user is shown several pieces of content, selects one. The system pops up a dialog box asking if the user was happy with the choices.
- A self-driving car takes a user to their destination, and as the user is getting out it asks if the user felt safe and comfortable during the trip.