Machine Learning Problem (Narrow it Down)

A machine learning problem should be narrowed down to as specific as possible. It's definitely not a bad thing to describe a machine learning problem generally or making it verbally appealing or having an ambitious problem definition is not bad either, at least to make the general people understand the problem better. But you also have to consider that machines don't understand things the same way we do.

Sometimes we humans understand something intuitively. But for machines the process of learning is nothing but a well defined algorithm (designed to solve specific kinds of problems) and using it to extract patterns and insights from the data. And that's why it's important to have a clear problem definition that can be divided into specific steps or tasks so that the machine understands what it's doing.

A machine learning problem should have a specific measurable goal that you can break down into specific actions in order to reach it.

For any kinds of business problem we have to narrow it down to a specific question in order to use a machine learning model to solve our problem. Just like most areas of life, you can't get good results if you don't start with a good question.

For example let's say, you're part of a government department, and you want to make the process of deciding on contractor bids faster, because you're finding that this takes too long and uses too many human resources.

This is a good overarching goal, but again, much too broad. So narrow it down, look at the current processes, and see where the pain points are.

Say, every bid has to be painstakingly checked for internal consistency, and then it is compared against the other bids.

So your first step isn't actually which contractor do I use, it's something to do with confirming consistency. Likely, your first step isn't even a machine learning project, but an overhaul of the forms used, so they're consistent and to the point.

All right. Pretend we're at that point, and we have nicely designed forms that contractors fill out. You still want to check for the reasonableness of bids.

To frame this as a supervised learning problem, we need correctly labelled examples. If you have a history of flagging suspicious bids, you could use that to build a model to answer, does this bid need to be flagged? Or you could use the history of accepted bids, and go further, look at successfully completed contracts, and then your question is, is this contract likely to succeed? Keep in mind, you're going to want to pay close attention to the learning data you use. You'll probably want to combine the form data with other data available on the contractor, publicly available datasets or data from private government databases.

So the first step in your new improved bid evaluation process becomes the automated check.

▼ Related Courses/ Tutorials

- Machine Learning: Algorithms in the Real World Specialization, C01 - Introduction to Applied Machine Learning