Hyperparameter Designer

Jerich Lee

December 07, 2024

I've been thinking a lot about how cool/crazy it is to be able to make abstractions —*mappings*—to things that pop up frequently. There's this idea called *chunking* that I've noticed in a lot of things that I do on a daily basis—e.g., Rubik's cubing, guitar, mathematics. It's cool that once you are comfortable with a finite set of "things", you can combine these "things" in sequences to make new "things", transform these "things" (inversion, rotation, translation).

In mathematics, one would call these *linearly independent vectors*, and neat results are just elements of the span of this space. In Rubik's cubing, you start out learning the *sexy move*. Once you know the sexy move, you learn the inverse, then the lefty variants of it. It turns out that the *sexy move* is just an element of the class known as "triggers". Once one learns these triggers, they allow you to "unlock" an entire new set of algorithms using just linear combinations of these triggers. This is how speedcubers can learn and retain thousands of algs—they aren't actually learning anything new—they are just learning new ways to combine things that they already understand deeply. It's like how Americans (and a whole lot of other people) learn 26 letters—then these letters unlocks the door to speech, reading, etc. In guitar, if you learn a few scales (these scales are our sets of "things"), you can improvise to any song—all you have to do is apply the scale functions to the right key.

As an engineer (student), one of the most frequent sets of "things" that I use is applied mathematics. In applied mathematics, you basically start seeing the physical world as combinations of mathematical functions—which are our sets of "things" in this context. This framework of modeling the real world not only allows me to reduce the real world to its essential bits, but also grants me the ability to communicate this information to other people—mathematics provides a lean framework, and also one that can be understood by others.

This had me thinking...what if discovery is just a combinatorics problem given a specific set of "things"? i.e., the enumeration of the space of all combinations of "things". If this is the case, what does it mean to design/create a set of "things"? Why does one set of "things" work better for a particular problem than others? What does it mean to change between sets of "things"? Does there exist some metric that allows one to capture the "distance" between two sets of things? The coolest thing is that using pure mathematics, we can answer these sorts of questions.

At the end of the day, something that I realized about being really comfortable with sets of "things" (I feel like I should probably find another way to state this) actually makes things really fun. There is so much enjoyment out of being confident and masterful of a

set of "things". Enumerating through this space is what people do—whether that is being really good at solving the top 75 LeetCode problems, getting really good at a particular super smash bros character, knowing how to prepare a certain Starbucks beverage—each space of things in each context becomes a playground for its users.