Adjective Sub-Types

by Sven Nilsen, 2018

A part of path semantics is developing syntax for making equations easier to read. An adjective sub-type is a way to describe objects closer to natural language. The motivation to make granular judgements use a form that can be both interpreted formally and naturally.

In general, an adjective sub-type if a chain of functions `$g_0$`, `$g_1$` added upfront of another function `f`:

$$x : g_0 \ g_1 \ f$$

$$f : A \to B$$

The object passed to a function is either the output of the function to the right, or if the type does not matches, object is the output of the function which output is input to the function the right. When there is no function to the right, the object is on the left side of `:`. With other words, when there is a function in front of another, it might "leap-frog" the function to the right until it finds an output that matches its input type.

If `$g_0 : B \to bool$` it means:

$$x : [f] \ [g_0] \ true \land [f] \ [g_1] \ true$$

If `$g_0 : A \to bool$` it means:

$$x : [g_0] \ true \land [f] \ [g_1] \ true$$

For example:

$$(a, b) : even \ len \ concat$$

There are two lists `a` and `b` such that when concatenated, the length is even. This means they have both even or both odd lengths.

Another example:

$$a : all\_positive \ even \ len$$

$$all\_positive : list \to bool$$

The `all_positive` function takes a list and leap-frogs `even` and `len` until its finds `a` that matches the input.

So far it might seem not very useful to use this syntax at all. It is harder to parse and looks weird when using it in many places. It is when you do analysis of ethics as rational reasoning with granular judgements about the world that it becomes useful, since natural language is a source of knowledge.

Assume we want to break down what the following sentence means:

> The blue vase is very nice.

When people say "this is very nice", they are not speaking of the information that is contained directly within the object. They talk about how this information is related to the rest of the world.

A granular judgement is how one thinks about a world given a statement as the only knowledge:

> E(x : [vase] [blue] true) = "very nice"
>
> x : world
> vase : world → object
> blue : object → bool

Here, `vase` is a function that the agent making the judgement uses to bring an object into attention.

The point is to *feel* about the blue vase as if it defined a subjective multi-verse relative to it: A large number of possible worlds which have in common that the blue vase exists. It does not matter whether good or bad things happens at the same time this *feeling* occurs, because high frequency events does not change the overall *feeling* that the blue vase gives.

While this might seems like an inconsistent way of navigating the world, it helps agents who reason rationally to focus on a few problems at a time. They judge the world as if their current problem they are working on was the only thing that matters, such they can engage and focus enough to increase the probability of solving the problem.

Notice that `E` is returning "very nice" which could be thought of as a function the following way:

> x : very_nice blue vase
>
> very_nice : world → bool

The `very_nice` function leap-frogs the `blue` and `vase` function.

Notice the similarity with natural language:

> This is a very nice blue vase.