Semantic Complexity of Binary Relations

by Sven Nilsen, 2020

In this paper I describe some formulas for computing semantic complexity of binary relations.

The semantic complexity of a binary relation algebra between `n` objects is measured in the number of possible interpretations of relations for the worst case scenario, when all binary relations are unlabeled.

For example, for `n` objects, there are `$n^2$` unlabeled binary relations. In the worst-case scenario, each of these binary relations are given a unique interpretation. Therefore, the semantic complexity is `$n^2$`.

The semantic complexity is a representative number for a group of complexity. This means that one can increase the complexity by allowing more interpretations of relations than there are binary relations between objects. Therefore, semantic complexity should not be taken as an absolute measure of complexity, but a relative measure.

By adding constraints for the semantic interpretation of binary relations, one can reduce semantic complexity. This can be encoded using modal logic, using `□` and `¬◇`, to handle absence of relations.

For example, if `(a, b)` has the same meaning as `(b, a)`, then the semantic complexity is:

∴	$n \cdot (n - 1) / 2 + n$	symmetric	∀ a, b { □((a, b) = (b, a)) }
∵	$n \cdot (n - 1) / 2$	pairs	∀ a, b { □((a, b) = (b, a)) ∧ ¬◇(a, a) }
∵	n	diagonal	∀ a { □(a, a) }

One can visualize this as coloring pixels in a `n × n` image, using as many colors as possible.

Constraint ∀ a, b, c, p	**Semantic complexity**	**Description**
-	n^2	No constraints
□((a, b) = (b, a))	$n \cdot (n - 1) / 2 + n$	Symmetric
□((a, b) = (b, a)) ∧ ¬◇(a, a)	$n \cdot (n - 1) / 2$	Pairs
□(a, a)	n	Diagonal
□((a, b) = (c, b))	n	Universal
□(p(a, b) = p(a))	n^2	Unique
□((a, b) = (c, b)) ∧ □(p(a, b) = p(a))	n	Unique universal

Notice that unique binary relations does not reduce semantic complexity, because more “colors” can always be added, more than enough to determine the second object. However, universal binary relations reduces the complexity by restricting the number of “colors” to `n`.