

From Mechanistic to Compositional Interpretability



AUTOMATING INTERPRETABILITY AS A CONSTRAINED OPTIMISATION PROBLEM...

Requires a measurable blueprint for discovering and evaluating mechanistic explanations.

Aim

Mechanistic interpretability aims to reverse-engineer neural computation into human-understandable parts in three steps: decompose, describe, validate (Sharkey 2025).

Gap

Without a formal framework, explanations can't be objectively verified, compared or composed. They can be locally valid yet globally inconsistent (Joshi 2026).

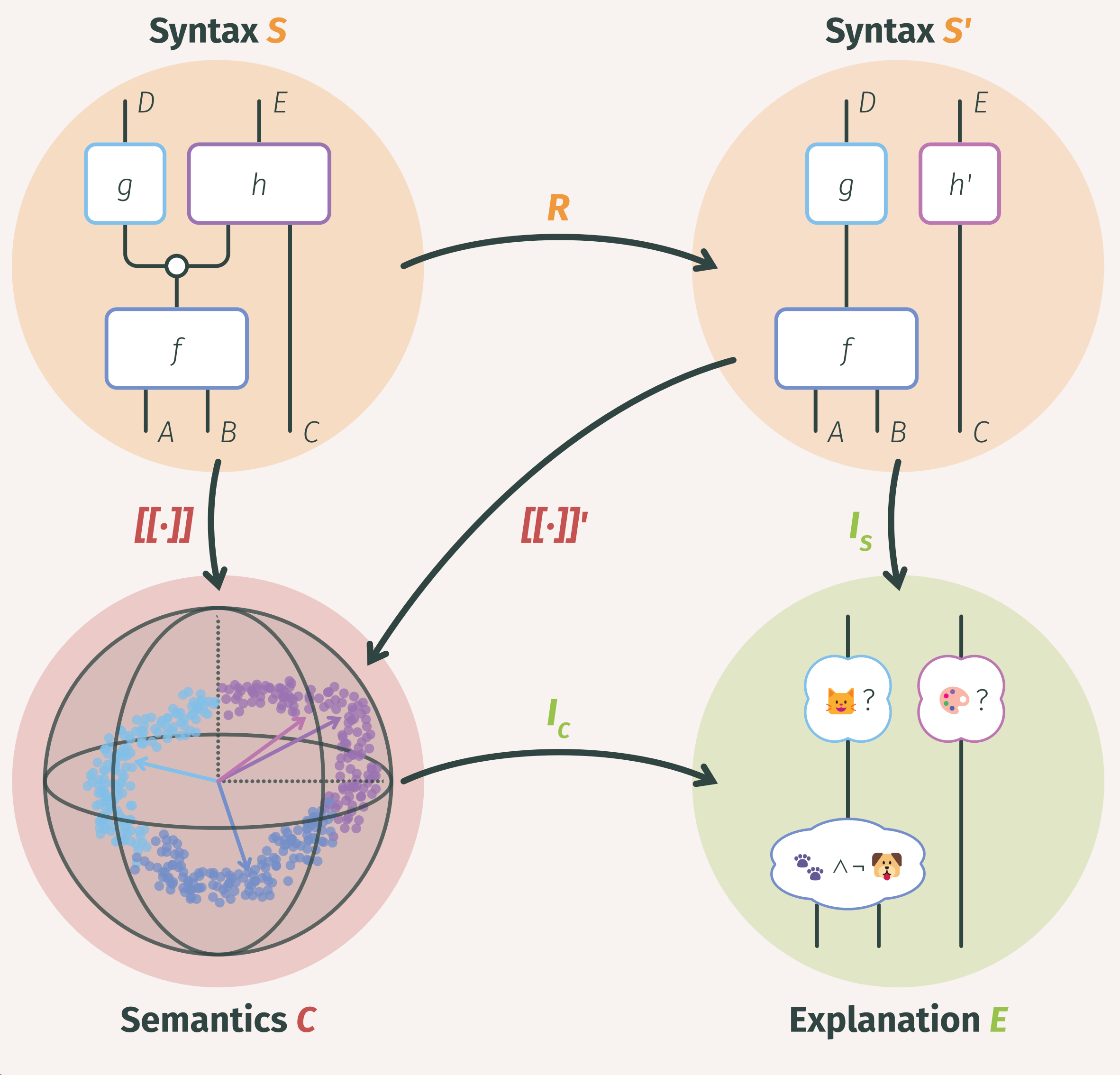
Framework

Compositional interpretability is based on two principles:

- Compositionality for grounding (category theory)
- MDL for selection (information theory)

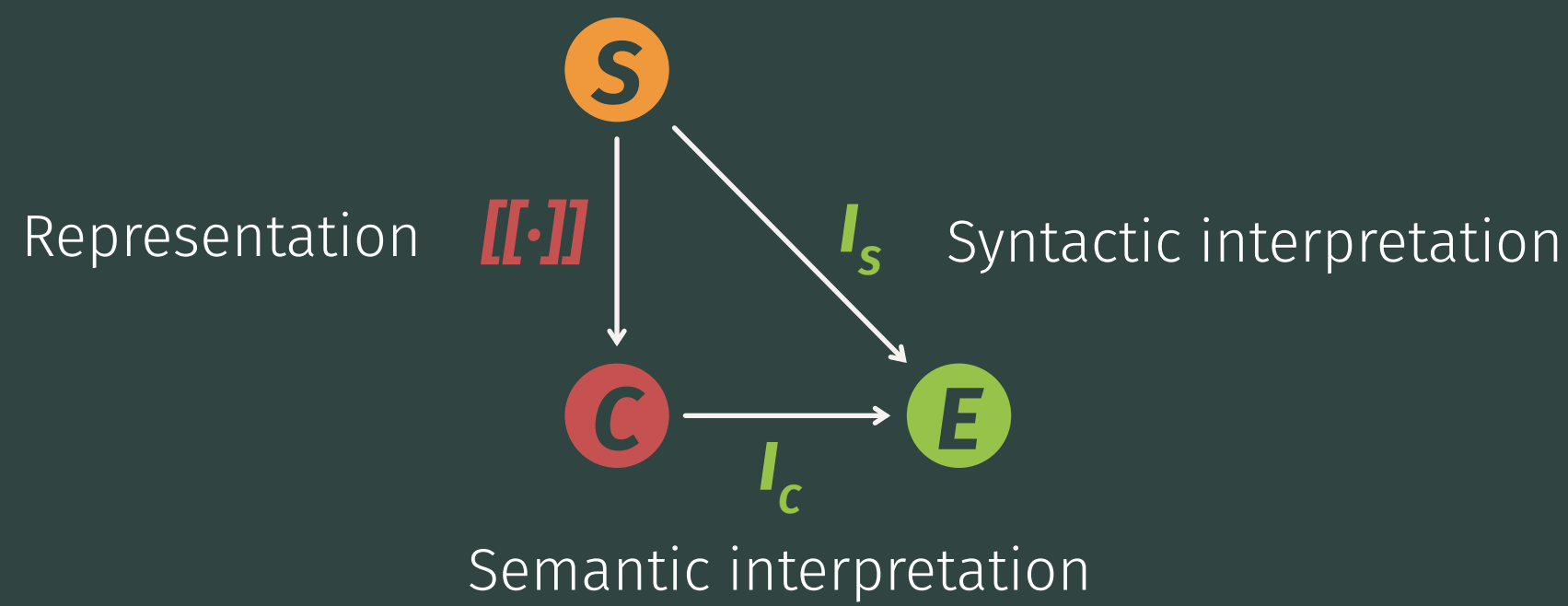
Payoff

- Formal criteria for explanation selection
- Spectrum of techniques generalising current methods
- Interpretability as a rate-distortion problem



1. WHAT ARE COMPOSITIONAL INTERPRETATIONS?

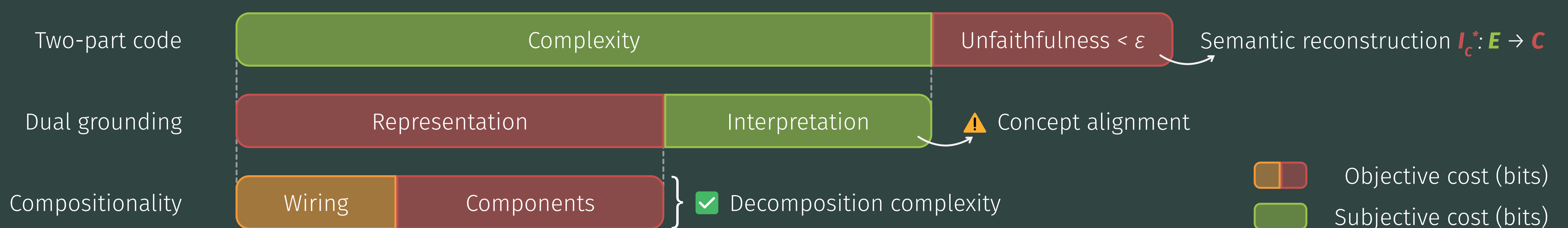
Model decompositions have *syntax* and *semantics*. Mechanistic explanations should be grounded in both.



- Syntax **S** E.g. string diagrams, feature circuits, PyTorch modules
- Semantics **C** E.g. differentiable functions, distributions, relations
- Explanation **E** E.g. natural language, causal models, pseudocode

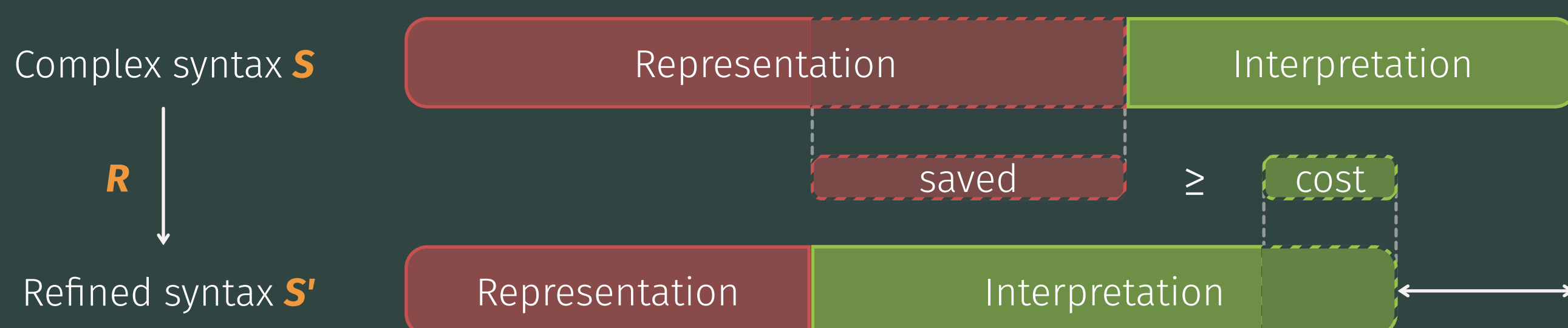
2. WHAT MAKES EXPLANATIONS GOOD?

Interpretation is communication. Good explanations have *minimal description length*. We give a measurable upper bound.



3. HOW DO WE FIND BETTER EXPLANATIONS?

Compressive refinement simplifies representations. Under the *parsimony criterion*, simpler means more interpretable.



Comonotonic coding suffices.

If the coding distribution encodes *explanatory optimism* (Ayonrinde 2025), both complexities fall together. More realistic codes give better refinements.

EVERY MECHANISTIC METHOD IS A COMPRESSIVE REFINEMENT...

They can be compared qualitatively, their explanations selected quantitatively, and judged by the parsimony criterion.

