Computational approaches to analogical reasoning

Background

• AI research recognizes importance of analogical reasoning in a wide range of human activities

• Main interest for us: can the computational approaches contribute to our understanding of analogical reasoning as used in the sciences?

• Ideally, each AI program is based on a computational theory or model of analogy: a set of assumptions, precise enough to implement in a computer program, about how the process of analogical reasoning does or should take place.
  - restricted aspects of analogical reasoning
  - many programs work on only a handful of carefully chosen examples.
  - trade-off between scope and ability to exploit detailed knowledge about the problem areas
Classification

- Most computational theories of analogy fall into two camps: *structuralist* and *case-based* views. Both approaches attempt to model the way humans use known solutions to solve new problems.

A. **Structuralist view:** analogical reasoning is essentially *syntactical*.
   - Analogies founded on parallel representations of knowledge about two domains.
   - ideal analogy: isomorphism. In general, value of an analogy is measured by the degree to which it approximates isomorphism.

B. **Case-based view:** analogies derive from the perception that two domains share salient or relevant dimensions.
   - these are characteristics we know to have been associated in the past with outcomes of interest, or to have interesting links to those outcomes.
   - The strength of an analogy depends upon the extent of relevant overlap between the source and target domains.

Q: Are both based on a deductivist model of analogical reasoning?
Three main differences:

- **Scope.** Structuralist programs tend to be *general-purpose* analogical reasoners. They can find and utilize analogies in virtually any context.

  By contrast, case-based reasoners are typically *narrow*; they work within a bounded ‘region’ of fairly similar cases. The most important features can be adequately described using a *finite and pre-established vocabulary*.

- **Flexibility of knowledge representation.** Structuralist programs employ a *general-purpose* representation of data about the source and target domains, such as some form of predicate calculus. Typically, there are many ways to describe the two domains in the formal language, and few guidelines for choosing between different representations.

  The main alternative, used in case-based reasoning (CBR), is to represent domains, or cases, with a set of *stereotypes* that provide a uniform way to store information. Stereotypes are typically implemented in data structures known as *scripts* or *frames*. Choosing stereotypes ensures a basis for comparing cases; however, it limits the scope of a program.

- **Treatment of Relevance.** On the structuralist approach, the relevance of a predicate or function is determined by *systematicity*: the extent to which it enters into complex networks of relationships. By comparison, CBR programs are oriented around a set of *indices* that includes every factor deemed to be relevant.
**Evaluation**

- Focus on potential for illuminating role of analogies in plausibility reasoning in the sciences.

1) **Predictiveness:** extent to which a computational theory makes definite predictions about which analogies are more plausible than others.
   - clarity and rigor of computational theories: virtues in their favour?
   - vagueness about input and knowledge representation
   - Predictiveness concerns both the evaluation criteria embodied in the program and the conventions about knowledge representation

2) **Applicability:** extent to which the evaluation criteria used by a program are justified.
   - reasonable procedures to discriminate between good and bad analogical reasoning (not just success in hand-picked cases)
   - Three aspects: definition of standards; implementation of these standards in the program or representation conventions; a philosophical argument that the programmer’s definition is reasonable.

3) **Scope or generality:** extent of the class of problems to which the theory is applicable.
   - does the theory account for diverse phenomena associated with analogical reasoning?
   - able to handle a large variety of analogies?

4) **Simplicity:** ability to account for many different types of analogies using relatively little conceptual apparatus.
   Since analogical reasoning is a complex phenomenon, however, the importance of simplicity should not be exaggerated.

**Comments:**

i) Virtues familiar from philosophical discussions of choice between scientific theories (e.g., Kuhn’s *Values, Objectivity and Theory Choice*).

ii) Trade-off between predictiveness/applicability and scope. CBR programs more reliable, but highly specialized. General-purpose, structurally based analogical reasoners rely on unspecified conventions about representation, so that predictiveness and applicability are problematic.