Dustin Jones

Ashok Goel

CS-7637

2016-09-25

Project 1 Reflection

While the final goal of the Project 3 is to build an algorithm that can solve Raven's Progressive Matrices (RPM) using strictly visual information, approaching first from a verbal perspective allows for a focus on the algorithms and techniques outlined by Knowledge-Based Artificial Intelligence: Cognitive Systems (KBAI) rather than computer vision solutions. The general approach of this agent is to break the overall task of solving RPM into several smaller tasks, each solving a segment of the problem.

Reasoning and Process

At its core, RPM are simply a collection of shapes, each shape with its own attributes. From the design of the problem, there is an assumption of relationships between each of the shapes, both within each figure and between figures. The first steps of this agent is to build a complete set of possible relationships between each individual object between figures. Each object has 2 possible states: either there is an object that it corresponds with, or there is not an object that it corresponds with. By making this assumption, the agent is able to assign a metric corresponding to the difference in number of shapes within each figure to help determine the likelihood of this particular relationship being correct. Once this list of potential relationships between objects has been established the agent calculates a new metric corresponding to the differences between those objects by comparing their attributes. With these two metrics combined, the agent can begin to rank each potential relationship for correctness.

Ranking the likelihood of each object pairing for correctness in isolation is not enough to solve the problem. Each pairing must be analyzed across all objects, first selecting the absolute best pairing by metric, then selecting the best pairings of the remaining objects.

For example, in Fig. 1 the agent first determines that there are 3 objects in A, and only 2 objects in B, therefore there must be a deletion. The large circle in A is paired with the large circle in B; because there are no changes, the metrics rank it first. Next, the small square in A is paired with the large square in B both because it cannot be matched with the large circle in B because that object is already paired with another object, and because there are fewer differences between the square in A and the square in B than there are between the small filled circle in A and the square in B. Finally, the agent determines that the small filled circle in A must be deleted because there are no remaining unpaired objects in B.

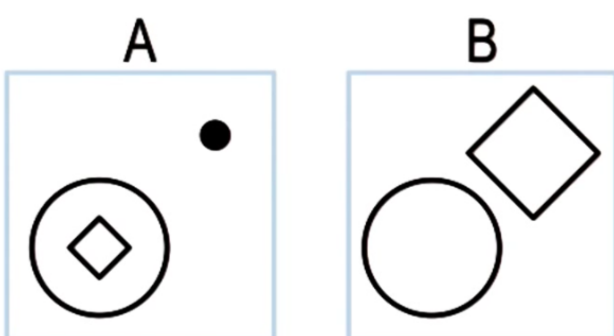

Fig. 1. Example of 2x1 matrix

Once the best solution of object to object mappings has been identified, the transformations between the objects are then applied to the corresponding figure. That is to say that the transformations from A to B are applied onto C to D, or if that particular set of transformations fails to apply due to object addition/deletion, then the transformations from A to

C are applied onto B to D. Finally, the resultant objects in the calculated figure are compared against the provided solutions to check for a match using the same object to object mapping logic as discussed above. If no exact matches are found, a final attempt is made to determine if a match exists by applying the above process to the provided answers to try and calculate a figure that matches figure A. If this step fails, then no attempt is made to 'guess' at an answer and the problem is simply skipped.

Risks, Mistakes, and Improvements

Given the constraints of time and resources, both for the development of the agent and for the processing by the agent, there a few risks (potentially resulting in incorrectly calculated figures) with the current implementation of the agent:

1. Only attempting one set of transformations. Transformations are determined by directly comparing individual attributes. If for example a shape's size is transformed from medium to large, the transformation is identified as 'make the shape large', however the actual transformation might be 'increase the shape's size by 1 step'.
2. Not directly relating objects through relational attributes. Relational attributes are compared by simply matching the number of objects within each attribute. For example, if a circle is inside both a triangle and a square, the agent simply 'sees' that the circle is inside 2 things, but not specifically which 2 things.
3. Not accounting for transformations which result in additional related objects during the object mapping process. Objects are mapped on a one-to-one basis, not on a one-to-many basis. For example, if there is a circle in one figure, and 2 identical but differently located circles in the next figure (as in a duplication), one shape is selected

as the object's map, and the other is simply viewed as an addition with no relation to the original.

4. Looking for an exact match. When a figure is calculated and subsequently compared against potential solutions, only an exact match will be identified as an allowable answer.

While taking these risks may have allowed the agent to be built both more simply and processing more quickly, with apparent short term success, they may prove to be an encumbrance when the agent is adapted to process 3x3 RPM. The agent most certainly could be adapted to account for Risk 1, potentially increasing the percentage of correct matches when evaluated with a larger problem space. Risk 2 will be significantly more challenging to address as it could easily result in combinatorial explosion when dealing with more than a handful of objects, however this particular risk may not result in significant gains in matching as there are many other attributes that may match sufficiently to derive calculated figures. Risk 3 will be challenging to address, but it also may be relatively unimportant it could already be accounted for through simple add/delete object transformations. Risk 4, while having the effect of virtually eliminating incorrect answers, also doesn't allow for guessing when there may be an answer that is 'close enough'. By adding the ability to select an approximate answer, the agent may be able to re-evaluate its process to account for unknown transformations, add it to its library of transformations and apply it to future problems.

KBAI Methods and Human Cognition

While the approach for the design of the agent is mainly object-oriented in nature, there are underpinnings of KBAI used to strengthen the design. The mappings between objects and their associated transformations are at their most fundamental levels, a semantic network. The potential mappings are identified through Generate & Test methodologies by building a complete set of all possible object mappings, creating a metric through which to adjudicate which mapping is the best mapping. Frames are used to identify what transformations are possible to apply to which objects and attributes.

Even though these pieces of KBAI are used to construct the agent, they still do not amount to an AI just yet. These pieces combined make up much of the reasoning portion of the deliberation process. The agent however still has no ability to directly learn or infer new transformations or relationships even if it has all possible information available about the current problem, it also has no ability to recall information about past problems. Eventually through use of new KBAI techniques and their related data structures the agent may be able to become an autonomous learning system.

This particular revision of the agent uses a very similar pattern of problem solving when compared to a human approach. When a human encounters a RPM, they do not simply try to solve it all at once, but break the problem into pieces: identify which shapes match which, what happened to them and in what ways can we project what happened to one set of shapes onto another. In some ways the agent can be more efficient that their human counterparts by ensuring that all mappings are evaluated quantitatively, rather than potentially erroneously ignoring mappings and transformations that may in fact be the correct case. Humans however currently have the advantage of easily being able to 'short circuit' an evaluation path if they can see that it is not going to arrive at a valid conclusion.

Works Cited

Carpenter, Patricia A., Marcel A. Just, and Peter Shell. "What One Intelligence Test Measures: A Theoretical Account of the Processing in the Raven Progressive Matrices Test." *Psychological Review* 97.3 (1990): 404-31. Web. 23 Sept. 2016.