Scott Dickerson
rdickerson3@gatech.edu

Generate and Test

How would you use **Generate & Test** to design an agent that can answer Raven's progressive matrices?

Generate and Test is a problem solving method. When coupled with a knowledge representation, such as Semantic Networks, it can be applied to a problem in Ravens Progressive Matrices to determine the optimal solution in the solution set.

The method can be applied by using a known transformation to generate a potential solution which is tested by comparing it to each solution in the solution set. Another way it can be applied is to insert each solution in the solution set as the problem solution and generate the missing transformation. This generated transformation is tested against the transformation from the given set of figures.

When testing potential solutions, context specific weighting rules will be applied by the tester in calculating a weighted score to each solution. Once all potential solutions have been tested the solution with the highest score is chosen as the optimal solution if it's score meets context specific efficacy rating. If no solution meets the efficacy rating, the opposite approach could be applied. If still no test passes the efficacy threshold the tester could settle for the highest score to determine the optimal solution or attempt to generate an alternate transformation for the given set which would result in a different generated solution to compare to the solution set.

Dumb generators and testers provide a very basic level of generating possible solutions and testing those solutions. A dumb generator would generate all possible solutions without consideration for the efficacy of each solution. A dumb tester would not dismiss any solution that passed some basic test (i.e. solution figure contains the expected number of objects).

With smart generators the knowledge of the concept being tested becomes more fine-grained which leads to fewer generated solutions to test. With smart testers more of the generated solutions will be discarded based on its more in depth knowledge of the concept as well as ability to identify solutions that can not possibly be the correct answer.

In general Generate and Test can be applied to the problems in Ravens Progressive Matrices as follows:

1. Select node in semantic network representing the source figure.
2. Select node in semantic network representing the target figure.
3. Compare the objects in the source figure with objects in the target figure and produce a transformation which is a set of transformation attributes (i.e. object x rotated).
4. Select node in semantic network representing the source figure from set with missing figure.
5. Apply the known transformation to selected node to produce new node with generated solution.
6. Compare generated solution to each solution in the solution set giving each solution a score that indicates how closely it matches the generated solution.
7. If the solution with the highest score has a score that exceeds the agent's minimum efficacy rating (i.e. scores 95 and min efficacy rating is 90), pick this solution as the correct answer.
8. If no solution meets the minimum efficacy rating, select node in semantic network representing a solution from the solution set.
9. Generate the transformation from the source figure from set with missing figure to the solution.
10. Compare this generated transformation to the transformation created in step 3 giving this transformation a score indicating how closely it matches the transformation from step 3.
11. Continue this process for all solutions in the solution set.
12. If the transformation with the highest score has a score that exceeds the agent's minimum efficacy rating, pick the corresponding solution as the correct answer.
13. If not, pick the solution from either step 7 or step 12 as the correct answer.