META COGNITION AND META REASONING

Introduction:

If one has to imagine a world where humans and AI agents can co-exist, in such a world the AI agents must be able to provide explanations of their actions. When creating a human like intelligence, designing and embedding reasoning and reasoning about reasoning must be an important factor to be considered. Only when AI agent can reason their actions and can explain them, humans can trust the agents. Only with that trust can humans and AI agent can co-exist. Explanations can be provided by reasoning about reasoning and this process of reasoning one's own actions and reasoning is called meta reasoning. Meta reasoning can be embedded in Knowledge based artificial intelligent agents. This is possible because of the explicit knowledge representation in knowledge based artificial agents. Knowledge based artificial agents have knowledge as opposed to ML based or some other computational based artificial agents which have data. In this report, I will be discussing meta reasoning in an artificial agent that can solve raven's progressive matrices verbally and visually. I will be using the RPM used in figure 1 to address meta cognition and meta reasoning in artificial agents.

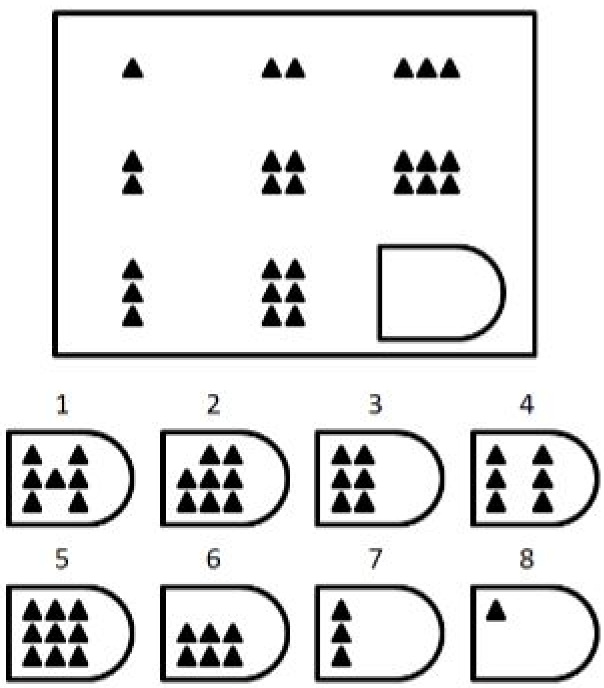


Figure 1

Human approach:

Let's analyze how a human (myself in this case) will solve the RPM from figure 1. I went through the rows and columns of the matrix and analyzed that from left to right and from top to bottom the objects count is increasing. So I will go through the options and eliminate 3,4,7,8 options

and now I will try to observe a deeper pattern so that I can converge to a single option. I observe that vertically from top to bottom and horizontally from left to right, the ratio of number of objects are in the ratio of 1:2:3. Now the missing element of the matrix must have 9 objects. By this logic I eliminate options 1,2 and pick option 5 as my solution.
When faced questions about my reasoning and solution, I will be able to provide explanation using the logic I used to arrive at the solution. Consider the questions mentioned in the problem statement:

(1) What is the object in the eight images displayed in the matrix?
I will say it's a triangle because the object in all the images matches my expectation of a triangle that a triangle must be a polygon with three sides.
(2) What is the relationship between the three images in the top row of the matrix?
The number of objects is increasing from left to right in the row and they are increasing by count 1 and in the ration 1:2:3.
(3) What is the relationship between the three images in the middle column?
The number of objects is increasing from top to bottom in the column and they are increasing in the ratio 1:2:3.
(4) What is the difference between the #3 and #4 as potential answers?
There is no difference. Both are not potential answers because in both the images the objects count is not increasing.
(5) Why did you select #5 as the answer?
Because number of number of objects are increasing in #5 and the objects count increase in #5 satisfies built 1:2:3 object count increase logic.
(6) Why is #3 not the correct answer?
Because the number of objects in #3 is not more than number of objects in F or H.

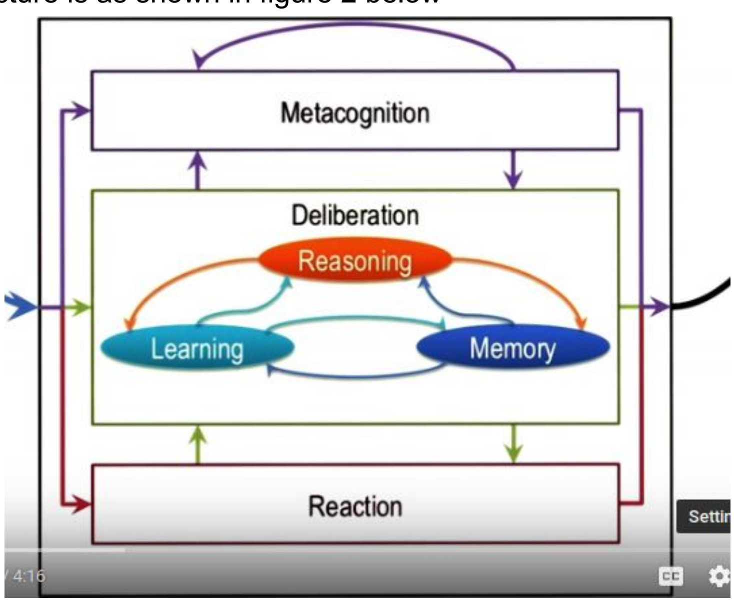
Agent's Architecture:
The agent's architecture is as shown in figure 2 below


Figure 2

Agents will have meta-cognition layer on top of deliberation layer. All the action taken in deliberation layer will guided by metacognition layer and the metacognition layer will get feedback from deliberation layer.

Agent's approach:

Knowledge representation in agents can be done using Frames or Scripts or Knowledge graphs. Now my agent should have a metacognitive layer that specifies depending on the purpose, a particular knowledge representation must be chosen. For example, my agent's metacognitive layer can be programmed in a way that if the current problem is a single instance i.e., it has no correlation with series of events coming up, then the agent should use frames to represent its knowledge. Else if the current knowledge is coming in form of a continuous stream, then the knowledge representation should be using scripts. Now the metacognitive layer also must be able to analyse whether or not the current action has sequences. Because my agent is solving RPM problems and every problem is independent of other problems presented, the agent chooses to represent knowledge in form of frames. If both verbal and visual data is present for a problem, then the meta cognition layer should help choose which data is to be chosen. Meta cognition layer should have a rule that says because verbal data is in more organized knowledge format, verbal data must be considered.

Verbal Data is represented as shown in figure 3.

At the deliberation level, agent will use verbal data and build knowledge in form of frames as shown in figure 4.

- Problem Name:
  - Problem Type
  - Has Visual
  - Has Verbal
  - Figures
    - Figure Name
    - Objects
      - Object Name
      - Object Attributes

Figure 3

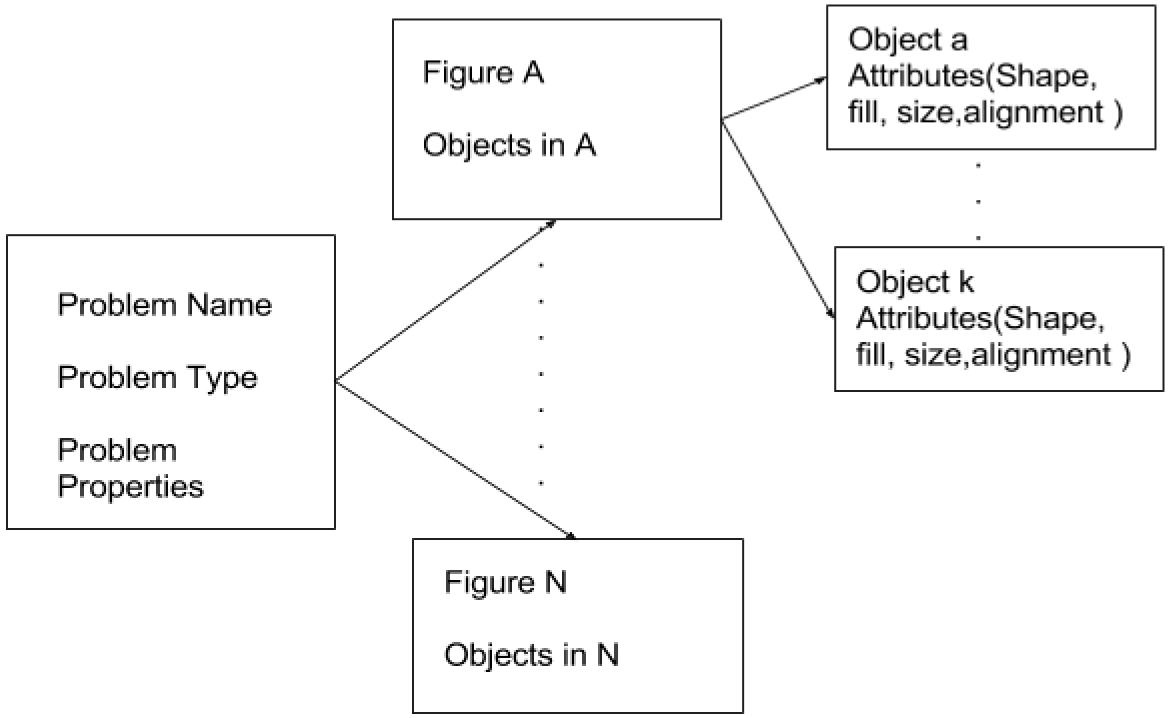


Figure 4

After extracting data and building the frames, the agent can choose one of the following methods to arrive at a solution

- Semantic networks and generate and test
- Case based reasoning
- Logic
- Analogical reasoning
- Production rules and systems
- Means end analysis

Now the agent has to decide which method to choose and metacognitive layer is going to decide. Meta cognitive layer will have rules that assign specific methods to specific rules. For example if there are no cases stored in the memory then case based reasoning method cannot be chosen.

Now for the current question assuming that cases are not getting stored in the memory, the agent cannot use case based reasoning and analogical reasoning. There must be another rule in meta cognitive layer that specifies: for visual data, semantic networks work best to represent object relations; Production rules is best method to represent changes occurring between two images; generate and test method is best way to choose an answer if provided with options. Means end analysis is best used if the result is known and the steps to get the results is to be determined.

Using the above meta cognitive information, the agent can choose semantic networks to represent relations, production rules to analyze the transformations and generate and test method to arrive at a solution. Thus meta cognition helps in strategy selection and integration.
And the algorithm for a 2x2 RPM which the agent follows will be as follows:

Algorithm:

The main **solve** function:

Read all the figures available. Find relation between lexicons of A and B i.e., to build a vocabulary to describe relation between lexicons and structurally represent the relation between A and B. Semantical identification and representation. Relate lexicons A and C. Built the relations and semantic networks between C and each option for D provided. Built the relations and semantic networks between B and options provided for D. Compare semantic network of A:B with C:{1,2,3,4,5,6}. Assign score for {1,2,3,4,5,6} depending on the match of C:{option} with A:B. Compare semantic network of A:C with B:{1,2,3,4,5,6}. Assign score for {1,2,3,4,5,6} depending on the match of B:{option} with A:C. The option with maximum score is selected as the answer

The **FindCorrespondingObjects** function: This function maps the objects between two images

Sort all the objects in A and B. For every object in A and B list all the attributes and save them in a dictionary. If number of objects in A is less than or equal to the number of objects in B then for every object in A I map some object in B depending on the similarity index between the objects. The similarity index using production rules is calculated as follows: If two objects have same number of attributes then similarity index between these two objects is increased by one. If any of the attributes of the two objects have same value then the similarity index is increased by one. Finally for an object in A the object in B with maximum similarity index is mapped as the corresponding object. If number of objects in A is less than number of objects in B, then all the objects in B to which none of the objects in A are mapped are indexed as ‘added’. Else if number of objects in B is more than number of objects in A, then for every object in B I map an object in A which has the highest similarity index with the object in B. If number of objects in A is more than number of objects in B, then all the objects in A to which none of the objects in B are mapped are indexed as ‘deleted’. Dictionary of corresponding elements is returned from the function

The **GetRelations/Generator** function: This function build the semantic network i.e., it describes the relational changes occurring between the two images and generates a solution depending on the relational changes occurring between the input images.

The dictionary returned from the FindCorrespondingObjects function is used as input to this function. For every key (object of A) in the dictionary, there exists a value(object of B). When the indices are not equal to ‘added’ or ‘deleted’ then the attributes of both the objects are compared and a relation is drawn. This is repeated for all the objects in figures A and B. The semantic network representation I am employing is in the form of a dictionary that stores the values of objects, the attributes of the objects and relative changes in the attribute value between figure A and B. This function returns a dictionary in the format mentioned above

The **CompareRelations/Tester** Function: This function is used to compare the relational/semantic changes between AtoB and Cto Fig{1,2,3,4,5,6}. This is the tester function that compares all the options available and chooses best option.

The dictionary of relations which is returned from the above GetRelations function is used as input to this function. This function compares the two semantic networks and gives a similarity score. The similarity score is computed as: for every object's attribute in semantic relation 1, if it matches with the corresponding object's attribute in semantic relation 2 then the similarity score is increased by 1. The score is returned to the Solve function where the object with maximum score is selected as answer

If the specified algorithm is applied to the problem presented in Figure 1, then agent initially analyses all the images A to H and 1 to 8 and builds frames for every image. If a question like what objects are placed in every image or how many objects are in image C is asked, the agent by referring to the frames will be be able to provide an answer.

Now the agent performs semantic analysis and build relation between objects. The agent will follow certain methodology to map objects and relate lexicons. If posed a question on why a certain object is mapped to a particular object and not to something else, the agent will answer the question by referring to the methodology and the specific reason as to why or why not an object fits the expectation.

Now to find out the transformation between images, the agent uses production rules. Rules are applied on images horizontally, vertically and diagonally. Initially by observing images A to C agent must have concluded that the number of objects increase by count 1 from left to right, the object count is in ratio 1:2:3. One might ask why it picked the particular rules? The answer would be because no other rules satisfy or can explain the transformations. The agent tries to apply same rules to middle row. Because the first rule that object count increases by 1 doesn't satisfy agent drops that rule. The agent using the object count increases in ratio 1:2:3, can find the missing image from the matrix using generate and test method. The generator will generate an expected answer with 9 objects as image G has three objects and object count increases in ratio 1:2:3. The tester will compare all the options with the generated expectation and eliminates all other options but 5 because only in the #5 image the object count is as expected. Now if faced with the questions stated, the agent using meta cognition will be able to explain its reasoning.

(1) What is the object the object in the eight images displayed in the matrix?

The agent using its knowledge base and frames must be able to answer that the object is a triangle.

(2) What is the relationship between the three images in the top row of the matrix?

Using the production rule agent picked, the agent must be able to answer that the number of objects is increasing from left to right in the row and in the ration 1:2:3.

(3) What is the relationship between the three images in the middle column?

Once again using the production rule, agent must be able to answer that the number of objects is increasing from top to bottom in the column and they are increasing in the ratio 1:2:3.

(4) What is the difference between the #3 and #4 as potential answers?

Both of them are not potential answers because both do not satisfy the chosen production rule.

(5) Why did you select #5 as the answer?

Because it satisfies the rule or pattern followed that the object count should be in ratio 1:2:3.
(6) Why is #3 not the correct answer?
Because it doesn't satisfy the rule followed that the object count growth should be in ratio 1:2:3.

Note that the agent is answering all the questions based on frames and production rules i.e., based on the it's knowledge. To be able to answer questions about its reasoning, agent's meat cognitive layer should be designed in a way that for every problem it encounters, the agent should maintain a case which should contain information about why a particular action is taken at a particular instance. For example when agent rejects #3 as potential answer, it must maintain a log in meta cognitive layer as in why #3 is rejected.
Meta reasoning can also help in rectifying the mistakes in reasoning and learning and filling the knowledge gaps. For example when a task fails the agent using metacognition can reason why the task failed. It can observe each and every step it performed and reason what might have gone wrong.
In relation to human cognition, you and me and every other human in the world has metacognition. We can reason about our actions and can produce explanations. To produce human like intelligence and to co-exist with humans, AI agent should be able to explain themselves. Only then humans can trust AI agent. Adding a meta cognitive layer to AI agents is a method to achieve this. Meta cognition is possible in AI agents because of the structure of knowledge base, memory organization and architectural design.