The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

The Structural Affinity Method for Solving the Raven's Progressive Matrices Test for Intelligence

Snejana Shegheva, Ashok Goel
Design & Intelligence Laboratory, School of Interactive Computing,
Georgia Institute of Technology
85 Fifth Street NW, Atlanta, GA 30332 USA
sshegheva3@gatech.edu, goel@cc.gatech.edu

Abstract

Graphical models offer techniques for capturing the structure of many problems in real-world domains and provide means for representation, interpretation, and inference. The modeling framework provides tools for discovering rules for solving problems by exploring structural relationships. We present the Structural Affinity method that uses graphical models for first learning and subsequently recognizing the pattern for solving problems on the Raven's Progressive Matrices Test of general human intelligence. Recently there has been considerable work on computational models of addressing the Raven's test using various representations ranging from fractals to symbolic structures. In contrast, our method uses Markov Random Fields parameterized by affinity factors to discover the structure in the geometric analogy problems and induce the rules of Carpenter et al.'s cognitive model of problem-solving on the Raven's Progressive Matrices Test. We provide a computational account that first learns the structure of a Raven's problem and then predicts the solution by computing the probability of the correct answer by recognizing patterns corresponding to Carpenter et al.'s rules. We demonstrate that the performance of our model on the Standard Raven Progressive Matrices is comparable with existing state of the art models.

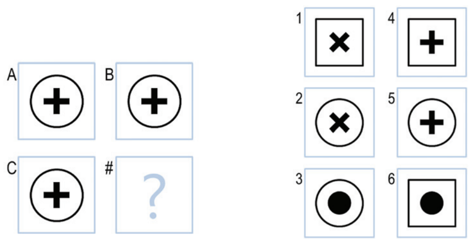
Introduction

Polya wrote that heuristic reasoning, or *ars inveniendi*, aims at discovering rules for solving problems for which an optimal solution may be impractical or requires a provisional plausible guess (Polya 1945). Newell has written that Polya's methods for problem-solving are directly relevant to mechanizing reasoning with computer programs (Newell 1983). However, heuristic reasoning traditionally has been used in conjunction with symbolic, propositional representations. With this work, we illustrate the use of heuristic reasoning with Markov Random Fields for addressing problems on tests of human intelligence. We aim at providing a complementary view on problem solving on the Raven's test that exploits statistical reasoning over graphical representations of the problems.

The need to be able to assess the degree of success for computational models of human cognitive processes has started an increasing trend of building computer systems capable of addressing tests of individual human intelligence (Bringsjord 2011). Hernández-Orallo et al. present an extensive taxonomy of about thirty existing models varying in purpose, generalization and technology (Hernández-Orallo et al. 2016). The diversity of of the problems ranges from geometric analogy (Evans 1964) to odd-one-out (Lovett, Lockwood, and Forbus 2008; McGreggor and Goel 2011) and Raven's Progressive Matrices (RPM) (Carpenter, Just, and Shell 1990; Lovett, Forbus, and Usher 2010; McGreggor and Goel 2014; Strannegård, Cirillo, and Ström 2013). In this paper, we center our attention on RPM because its visual input, variety of problems, centrality in previous research, and a correlation with general human intelligence provide a suitable dataset for analyzing capabilities of a computational model that focuses on problem solving.

A RPM problem is a clever organization of geometric figures (see Figure 1 for an illustration). In part because of its simplicity and partly because of the high correlation with other measures of intellectual achievement (Carpenter, Just, and Shell 1990), it is widely adopted in psychometrics, the science of measuring intelligence and knowledge. The Standard Raven Progressive Matrices (SPM) test consists of five sets of twelve problems each, A through E, with the problems typically increasing in difficulty both within a set and across the sets. In this discussion, we will focus on SPM problems (both, 2x2 and 3x3) that are presented in black ink on white background. The left-hand side of the Figure 1 shows a 2x2 matrix similar to a problem from the SPM test, with the missing element in the lower right corner. The right-hand side shows a set of six possible answers for filling in the blank cell to complete the logical pattern in the matrix.

Early computational models of addressing Raven's problems did not change the original problem representation, focusing instead on the rules needed to solve the problem. Carpenter et al, 1990, identified five distinct rules - "constant in a row", "distribution of three values", "quantitative pairwise progression", "figure addition", and "distribution of two values". Their method exemplified heuristic reasoning over propositional representations. The system proposed by Lovett et al. used prior knowledge of geometric elements and spatial relations to build qualitative spatial representations of the human generated sketches of RPM problems, and then performing analogical reasoning on them (Lovett



(a) RPM Problem Space (b) RPM Solution Space

Figure 1: Example 2x2 problem similar to one from the Standard Raven Progressive Matrices (SPM) test. (Due to copyright issues, all such figures in this paper illustrate problems *similar* to those on the SPM test.)

et al. 2009). These models illustrate the *Analytic* strategy to solving Raven's problems (Hunt 1974).

More recently, two new approaches, *fractal* and *affine*, propose purely iconic visual representations of the Raven's Progressive Matrices test (McGreggor, Kunda, and Goel 2014; Kunda, McGreggor, and Goel 2013). The *affine* method splits the matrix grid and the solutions grid into individual cells and performs affine transformations on the pixel representations. The *fractal* method takes this division further by partitioning the cells into fractal units and subsequently estimating similarity based on the features extracted from the fractal representations. Both approaches exemplify the *Gestalt* visual strategy, an alternative to the *Analytic* strategy (Hunt 1974). An important aspect of these approaches is that they operate on a *transformed representation* of the original problem (Hernández-Orallo et al. 2016).

A more recent model in the *Analytic* approach analyzes RPM problems in terms of their practical difficulty as measured by the number of rules applied in Carpenter et al.'s model (Ragni and Neubert 2014). A Bayesian model in this tradition assigns prior probabilities to the rules in Carpenter et al.'s model to fit data on human performance on RPM problems (Little, Lewandowsky, and Griffiths 2012). Another anthropomorphic cognitive model (Strannegård, Cirillo, and Ström 2013) emphasizes problem-solving strategies evident among high-achieving human problem solvers.

We observe two core themes common to the above computational accounts: problem re-representation and exploitation of problem-specific heuristic strategies. We present an alternative computational method that combines the benefit of purely visual representation, problem re-representation, structural mapping, and heuristic reasoning nested in the problem-solving strategies. We use the framework of Markov Random Fields (MRF) as the basis of the proposed mathematical representations because it enables us to express interactions between images in the RPM problems in a formal and very compact way. Markov Networks, which are a subclass of general graphical models, provide an advantageous mechanism for interpretation of the structure of the problem through assigning numerical values to interactions between its components.

Our Approach Motivation and Overview

The general process of our computational model is presented in Figure 2. The three main modules - *representation building*, *pattern learning* and *pattern recognizing* - constitute the essence of the Structural Affinity computational model for solving Raven's Progressive Matrices. The low-level pixel information is extracted from images and represented as affinity factors which measure the compatibility between images. This information is used to create the next level of abstraction - a problem structure corresponding to a rule that most likely captures the logical sequence in the input image. The learned abstractions are stored in memory and later retrieved during the process of pattern recognition.

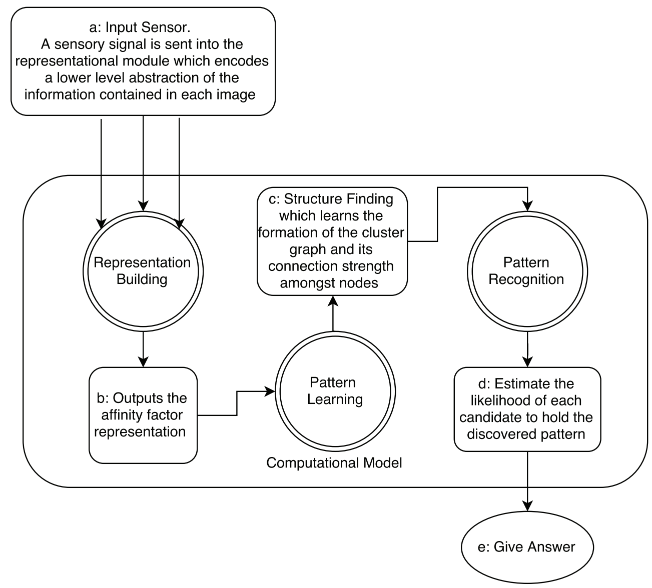


Figure 2: A diagram of the Structural Affinity computational model with three main modules - representation, pattern learning, and pattern recognizing.

We demonstrate the basic process on the example shown in Figure 3.

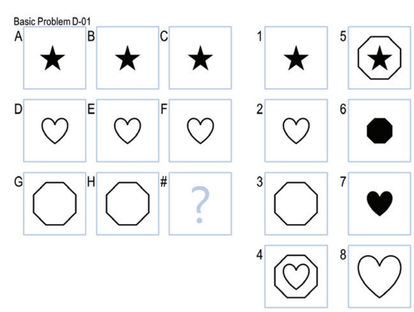


Figure 3: An example of the 3x3 Raven's problem

After receiving the image matrix (here, 3x3) and encoding the input with affinity factors, the pattern learning module starts the process of finding a structure which best encodes the spatial relationship between the images in the given problem. The search of a structure is initiated by analyzing the possible transformations when moving from left to right (row), or from top down (column). Given that there may exist more than one structure that can encode the problem with sufficient soundness, our methods gives preference to the formations which satisfy the following two principles of parsimony:

- Minimal number of the nodes: our method seeks the minimal group of images which together represent a single unit of a pattern. For example, for the problem given in Figure 3, the smallest group contains three images, and we could have a few different ways of grouping - row-wise, column-wise, diagonal or triangular
- Feature invariance: of all candidate groups, our method gives preference to those clusters which undergo the minimal number of detected transformations. For example, as the row grouping (for Figure 3) preserves the identity property better than the column grouping, the row-structure provides a higher information gain on the pattern.

After the minimal structure is mapped out, the pattern recognition module collects the evidence of a pattern similarity by fitting each of the candidate solutions from the given list. The task of the pattern recognizer is to find the solution with highest likelihood. By substituting each candidate into the incomplete third row, the recognizer scores the resulting formation with the objective to maximize the identity function. This process arrives at the correct solution by selecting the image that repeats the image observed in the third row - a circle. By being able to abstract the concept of a *rule* from the graphical representation of Raven's problem, the model learns to recognize a pattern. In the next section we show how to generalize this example by formalizing the method of exploiting the structure of the graph.

The Structural Affinity Method

Knowledge Representation

The core challenge in solving a Raven's Intelligence test is identifying the logical pattern in a sequence of geometrical images that when expanded optimally leads to a single correct answer. Thus, the issue here is how to establish a representational structure parameterized in such a way that it directly correlates with the underlying pattern?

Our approach here is to model the interactions between images as undirected graph structure such as Markov Random Field. Each *cell* of the Raven's Matrix corresponds to a *node* variable in a network with edges capturing the interaction between variables. At the crux of the method is the notion of *image affinity* which tracks an important measure for how compatible are the images within a group. The affinity between neighboring nodes is a *factor* function whose purpose is to parameterize the undirected graph without imposing a causal structure (Koller and Friedman 2009).

Let D be a set of N images. The affinity factor ϕ is then a function from $Val(D)$ to $\mathbb{R}$. As an illustration, let us consider the affinity factor for 2x2 Raven's matrix problems such as the one illustrated in Figure 1. The smallest building block of the image is the pixel which we shall denote as x^1 or x^0 for white and black colors respectively. Table 1 shows one potential factor function for a pair of images which requires four possible configurations of the pixels' color assignments.

		$\phi(A,B)$
a^0	b^0	1000
a^0	b^1	200
a^1	b^0	500
a^1	b^1	2000

Table 1: An example of the affinity factor ϕ for a pair of images A and B. It captures the interaction between variables by estimating the agreement between choice of pixel color. A high value for the assignments with matching pixel states, (a^0, b^0) and (a^1, b^1), correspond to a strong agreement, i.e. images A and B are very similar. On the other side, if the assignments with opposite pixel states, (a^0, b^1) and (a^1, b^0), are more likely indicated by a high ϕ value, then a significant transformation has been applied to image A to form an image B.

Formally, the factor function is defined as follows:

$$\phi(A, B) : Val(A, B) \mapsto \mathbb{R}^+ \tag{1}$$

The affinity factor is calculated by counting the number of occurrences for each configuration (a, b):

$$\phi_{(a,b)}(A, B) = \sum_{i=1}^{n-pixels} [C(A_i) = a] \wedge [C(B_i) = b] \tag{2}$$

where $C(X_k)$ is the operator function for reading color bit information for pixel k of the image X.

A full Markov Network for the 2x2 Raven's Matrix is graphically visualized in Figure 4 with nodes encoding image labels and edges parameterized by affinity factor functions.

For a 3x3 Raven's Matrix, the full graphical structure is more complex. However, we are mainly concerned with the factor function which takes a similar form with 2^3 possible assignments as indicated in Table 2.

			$\phi(A,B,C)$
a^0	b^0	c^0	ϕ_1
a^0	b^0	c^1	ϕ_2
a^0	b^1	c^0	ϕ_3
a^0	b^1	c^1	ϕ_4
a^1	b^0	c^0	ϕ_5
a^1	b^0	c^1	ϕ_6
a^1	b^1	c^0	ϕ_7
a^1	b^1	c^1	ϕ_8

Table 2: Affinity factor ϕ for a triple of images A, B and C in the 3x3 Raven's matrix problem illustrated in Figure 3

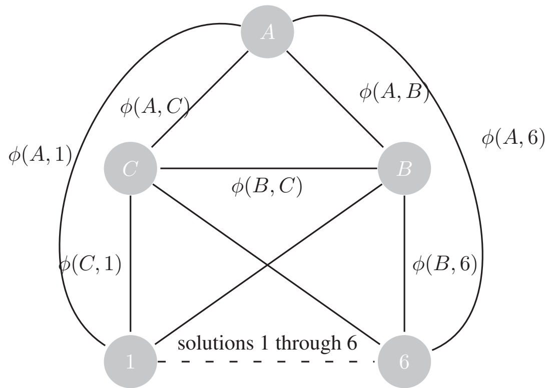


Figure 4: Markov Network for the 2x2 Raven's matrix problem as shown in Figure 1

Justification

Markov Networks are commonly used in research on computer vision for a variety of tasks such as image segmentation and object recognition (Koller and Friedman 2009). By formulating a model with the ability to capture the interactions between neighboring images in an RPM problem, we can infer the logical pattern in a sequence of images. This method of representing an RPM problem as a Markov Network not only does not involve any propositional representations (such as shapes, objects, spatial relations), it does not even engage any image transformations (such as reflections, rotations, translations). Instead, the reasoning is based solely on the statistical interaction between pixels in the images.

Rule Mapping - Pattern Learning

Solving Raven's problem using structural affinity method can be conceptualized as a process of going from a general model of the problem to a restricted Markov Network which captures the *minimal* amount of knowledge required for solving the problem. For the 3x3 RPM problem, the general model assumes inter-dependence of all components of the matrix space and requires $\frac{9(9-1)}{2} = 36$ edges which convolute the structure of the Raven's problem. To address this issue, we restrict the number of the edges to the minimal set of the most influential connections that reduce the problem complexity and aid in discovering a structure. Before unpacking the requirement for minimal knowledge, let us define the concept of structural affinity more formally.

Definition 0.1. Let $\mathbf{X}$ be a finite set of variables representing components in the visual problem and let $\phi_k(X_i, X_j)$ be a k-th factor function that denotes the affinity between X_i and X_j. We define the *Structural Affinity* $\Psi(\mathbf{X})$ to be a set of factor functions that contain the minimal representation of the dependence structure in the full graph G over given RPM:

$$\Psi(\mathbf{X}) : \{\phi_k(X_i, X_j) \,|\, X_i \not\perp X_j \; and \; k \leq ||G||\} \tag{3}$$

where $||G||$ is the cardinality of the graph G given by the total number of edges.

With a reduced set of affinity factors we represent a relationship between the components of the Raven's problem that highlights the strongest interactions in the network (i.e., images represented by the variables X_i and X_j that are not independent). The scale of dependence is captured by the affinity factor that measures the degree to which the images are compatible with each other.

For example, the images in the Raven's problem where the objects in a row simply repeat each other without any additional transformation, are said to be highly dependent resulting in large values of the affinity factor functions.

One approach of picking an informative network structure is starting with a variable of interest, in our case a candidate solution, and iteratively estimating its dependence on the neighboring variables. Initially, all variables in the matrix spaces are considered to be neighbors with the solution item. We then prune the weak dependencies leaving only the minimal map of the network structure which serves as a foundation for identifying a possible strategy for solving Raven's problem.

Structural Affinity Hypothesis The concept of a minimal map allows us to explore a possibility of identifying a correct solution of Raven's problem by merely analyzing its structure as a mapping of strength between its components.

> Given a Raven problem and its expected solution, the Markov network structure with the minimal map will have the highest structure score as compared to the networks with the sub-optimal solutions.

Under this hypothesis, solving Raven's test can be achieved with the following four steps of creating and selecting the most likely structure:

1. Create a hypothesis space Ω_m of all possible factor graphs, where m is number of possible solutions.
2. Define objective functions that optimize towards the desired properties of the graph G.
3. Compute the scores for all resulting factor graphs.
4. Select the structure with the highest score as a most likely solution to the problem.

Both, the *presence* and the *strength* of the edges in the created network, provide insight on the nature of the masked dependencies. The resulting network topologies highlight properties of Raven's problem which may be connected to the problem structure through the set of rules described in Carpenter et al.'s seminal cognitive model of problem solving on SPM (Carpenter, Just, and Shell 1990).

For example, a rule for *Figure Addition* (an object in a third row/column is formed from juxtaposition of the objects in the first two rows/columns) can be reduced to a network structure as shown in Figure 5 where a strong interaction is expressed by a presence of a solid edge, and a weak interaction is indicated by a dashed line. Little et al. refer to this rule as *Logical OR* to classify the transformation between figures as logical operations of disjunctions (Little, Lewandowsky, and Griffiths 2012).

Little et. al. show that a Bayesian model can successfully predict the rule by observing a collection of hand-coded fea-

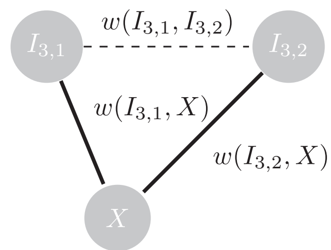


Figure 5: Network topology for the figure addition rule. The elements of the networks are images from either the third row or the third column, and the missing element X from the solution space. Here, the figures which are being added do not require dependence, however, the final figure is dependent of both constituents (dashed vs. solid edges)

tures. In our model, the handcrafting of features is not necessary as the rule is learned directly from the given problem. By following Little's et. al. Bayesian formulation, we compute the posterior probability of each rule using following formula:

$$P(g|s) \sim P(s|g) * P(g) \tag{4}$$

where $P(s|g)$ is the evidence for observing a structure s given rule g, and $P(g)$ is the prior for rule g. Here, $P(g|s)$ is an unnormalized measure since we use it for ranking purposes, and therefore do not need to convert it to probability value in the range [0..1].

We compute a rule likelihood by collecting evidence of the rule g from the observed structure s by performing a series of independence tests, such as χ^2, between affinity factors parameterizing a Markov network. The distribution of the χ^2 values within the network provides a basis for constraining functions $\mathcal{H}_i$ which after aggregation and normalization produce a score for the given combination of the structure s and rule g. The computed score measures the compatibility of the derived structure topology and the suggested rule:

$$P(s|g) = \frac{1}{Z} \sum_{k=1}^{n} \mathcal{H}_i(g) \tag{5}$$

where $P(s|g)$ is the likelihood of the structure s given rule g, Z is a normalization function, and $\mathcal{H}_i(g)$ is a constraining function of the form:

$$\mathcal{H}(g) = t(\bar{\chi^2}) \cdot w_t(g) \tag{6}$$

where t is statistics measure for the obtained χ^2 values of the independence tests and the $w_t(g)$ is the weight of the given statistics measure for the rule g.

$$w_t(g) = \begin{cases} > 0, & \text{for boosting the statistics} \\ 0, & \text{if statistics is not relevant for the rule g} \\ < 0 & \text{for penalizing the statistics} \end{cases} \tag{7}$$

Various weights and statistics are derived experimentally, based on the heuristics of the rule. For example, Carpenter et al.'s *constancy* rules give preference to the uniform distribution of the χ^2 values, i.e. all edges are connected with similar strength values, while *quantitative* rules penalize *evenness* and give preference to the configurations with a gradual decrease of strength.

Predicting the Response - Pattern Recognition

Once the rule tokens have been induced from Raven's matrix space for each problem, our method predicts the solution to the missing object as follows: For every inferred rule R_i for the problem at hand, rank each candidate solution X_j by computing the average value across a set of corresponding objective functions f_n imposed by the rule R_i.

$$E(X_j) = \frac{1}{||R||} \sum_{i=1}^{M} \sum_{n=1}^{N} f_n(R_i) \tag{8}$$

where $E(X_j)$ is the estimated likelihood of the solution S_j given set of objective functions f_n.

An example of an objective function for the *constancy* rule is:

$$maximize \quad F = \phi_{0,0}(I_1, I_2) \tag{9}$$

where I_1 and I_2 are variables in the Markov Network representing two images in Raven's problem affinity for which is being estimated with the factor ϕ. Here, the factor function is computed over states $(0, 0)$ (black pixels) to maximize the potential of identity property between images I_1 and I_2.

Finally, the solution is selected by simply finding the candidate with the maximal value of the likelihood:

$$X = \arg\max j \in [1..J] E(X_j) \tag{10}$$

The objective functions exemplify the heuristic reasoning of our approach by introducing conditions under which the most likely solution should be found. Given the statistical nature of the algorithm, the model combines the signals from an ensemble of heuristic functions to predict the solution that fits the observations represented in the form of affinity factors. The heuristics attempt to capture our expectations of the behavior between images in the Raven's test under the assumptions of the inferred rules. For example, by assuming that the best strategy for solving a specific Raven's problem is applying a conjunction rule in the row, we impose a kind of heuristics that simultaneously maximize the pairwise relationships between the candidate solution and the images from the third row. We achieve such heuristic constraint by applying the following objective functions:

$$max \quad F = \phi_{0,0,0}(I_1, I_2, I_3) \tag{11}$$

$$min \quad F = \phi_{0,1,0}(I_1, I_2, I_3) \tag{12}$$

$$min \quad F = \phi_{1,0,0}(I_1, I_2, I_3) \tag{13}$$

Initial inspection of the ranked candidate solutions indicated a strong propensity to split the predictions into two

Rule				
	B	C	D	E
Constant in a Row or Column	6	6	7	0
Distribution of three values	0	0	8	0
Distribution of two values	0	0	0	1
Figure Addition or Subtraction	3	5	1	21
Quantitative pairwise progression	2	14	3	0
Symmetry*	5	0	0	0

Table 3: Induced Carpenter's rule for set B, C, D and E. We additionally included rule *Symmetry* for set B which is not classified as one the five Carpenter's rules. This analysis included cases where multiple tokens is required for solving a problem.

groups separated by a large margin - highly unlikely candidates which when selected would probably be attributed to a random choice, and plausible candidates, the incorrect answers amongst which are typically due to *Incomplete Correlate* errors in human performance on the Raven's test(Kunda et al. 2016). Our technique currently does not directly measure the confidence, however, given the probabilistic nature of the approach, the confidence is inferable via the computed score for each candidate.

Results

Setup Description

We have evaluated our computational model for sets B through E of the Standard Raven's Progressive Matrices (SPM) test. This is because the five types of rules described by Carpenter et al. do not apply to set A which relies heavily on textures and not geometric pattern learning. Images were digitized for consumption by our computational model. The resulting artifact introduced some noise related to image alignment; however, the statistical nature of the algorithm *smoothed out* the impact on the accuracy.

Rule Inference Results

Table 3 reports the overall frequency of each inferred rule according to the Carpenter et al.'s naming scheme. The number of rule tokens applied to a problem varies from 1 to 4, with a most frequent case of 2 rules per problem. In addition to inducing the rule, the algorithm augments the results with the directional variable (row, column, diagonal, or triangular), however, for the analysis and comparison we only kept the name of the rule.

To estimate the correctness of the rule induction outcome, we compared the algorithm response to a suggested logical relation in the items for each problem in sets C, D, and E (Georgiev 2008). For set B (which was not included in the Georgiev's analysis), we manually created the ground truth for each problem before hand. Our performance measurements resulted in 0.94 for precision, 0.72 for recall and 0.82 for F1 score. The lower number for recall indicates that we did not retrieve all rule tokens, however, as we show the agent problem solving accuracy results below, it did not impact significantly the results generated during the solution prediction phase. We found that in most cases the rule tokens which were identified (94%) bore sufficient information for disambiguating the correct solution.

Set	Correct	Percentage Correct
Raven's Problem B	11	91.67
Raven's Problem C	12	100.0
Raven's Problem D	10	83.33
Raven's Problem E	11	91.67

Table 4: Response predictions for rule-aware agent per set for Raven's Standard Progressive Matrices Test - Set B, C, D and E

Rule-Aware Agent Results

Our computational model predicts correct responses for 44 out of 48 targeted problems (sets B through E of SPM) resulting in overall 91%. Table 4 shows the absolute count of the correct responses and its percentage per set.

The incorrect responses for the 4 out of 48 problems can be mapped to the two out of four conceptual error types - *Wrong Principle* and *Incomplete Correlate* in Kunda et al.'s classification of error on SPM (Kunda et al. 2016). As the Structural Affinity method is based on the image agreement, in a case of ambiguity it is biased towards selecting an answer which is a copy of the elements from the matrix space of the problem (*Wrong Principle*), or the answer is almost correct, i.e., second best choice (*Incomplete Correlate*).

Comparison to Other Computational Models

Table 5 shows the comparison of our technique based on Structural Affinity method with two visual methods - Affine (Kunda 2013) and Fractal (McGreggor and Goel 2014) and two propositional methods -Anthropomorphic (Cirillo and Ström 2010) and CogSketch (Lovett, Forbus, and Usher 2010). The Affine and the Fractal methods are using a visual approach and problem re-representation which relate to the Structural Affinity in Gestalt principle. The Anthropomorphic and the latest published CogSketch models differ from our method in the overall strategy, and as such also serve as good comparison models. The Structural Affinity method has a total score of 44 out of 48 targeted problems in the SPM; the Affine and Fractal have total scores 39 and 42 respectively on the same problem set. The Anthropomorphic model solved 28 out of 36 targeted problems (C through E), and, finally, the CogSketch reported solving 44 out of 48 problems. The results presented in our Structural Affinity method compare very well in performance with CogSketch and Fractal methods and outperform Affine and Anthropomorphic accounts. Thus, we suggest that the method of first learning the pattern corresponding to Carpenter et al.'s rules as tokens, and then recognizing the answers that fit the pattern the best, plays an important role in constructing a model capable of solving intelligence tests in the form of geometric analogies.

Method	Correct	Target
Affine	39	60
Antropomorphic	28	36
CogSketch	44	48
Fractal	42	60
Structural Affinity	44	48

Table 5: Comparison of the overall accuracy results for five computational models

Discussion and Conclusions

We began this paper by asserting the importance of learning and recognizing patterns in problem-solving by exploiting the interpretative nature of graphical models. We offered a representation that is capable of quantifying the level of interaction between constituents of the geometric problem on an example of Raven's intelligence test. The simulation results presented in this work show that our Structural Affinity method built on the basis on Markov Networks allows inducing the set of rules (pattern learning) that are most imaginable to express the logical sequence used for creating the problem. As our model generates input for structure finding algorithm directly from the images, it is not susceptible to the limitation of Carpenter et al. (Carpenter, Just, and Shell 1990) and Little et al. (Little, Lewandowsky, and Griffiths 2012) works where the model inputs are hand-coded.

The success of the rule inference portion of the algorithm (94% precision score) suggests that geometric problems on the Raven's test can be understood through statistical analyses that are also frequently leveraged in understanding other phenomena such as language and vision. The cases where the model misclassified the patterns are also difficult for humans as they require the analysis of slopes and completeness (Georgiev 2008). One of the problems, however, pointed to the limitation of our own approach where affinity factors could not capture the corresponding regions of the images. It may be possible to address this criticism by a more advanced factor-creation algorithm that can account for a more significant image misalignment.

The second component of our algorithm - applying the induced rule tokens to Raven's problem to predict the solution - demonstrates the heuristic nature of problem solving through pattern recognition. Each inferred rule dictates the set of heuristics expressed through affinity factor states' relationships either validating or refusing a solution (we showed an example of an identity heuristic function). Our achieved accuracy of 91% is on par with the state of the art computational accounts for solving SPM. The misclassified cases, most suitably attributed to a *Wrong Principle* or an *Incomplete Correlate*, suggest a potential improvement of the solution disambiguation by adding a more comprehensive set of heuristics. The generality of the model is supported by adhering to the standard set of rules as classified by Carpenter et al. (Carpenter, Just, and Shell 1990) and applied to Bayesian models and evaluated to human performance data fitness (Little, Lewandowsky, and Griffiths 2012).

Our method composed of three modules - representation building, pattern learning and pattern recognition - raises the question on the power of statistical reasoning for understanding problems that require logical deductions: what is minimal level of abstraction that is sufficient for pattern extraction and accurate response prediction? The evidence shared with the results of our simulations suggests that high accuracy levels are indeed achievable with statistical reasoning over graphical models. Furthermore, as the underlying process of identifying structure is not specific to SPM, our method may be more general and applicable to other visual problems that require deducing logical sequences.

Future Work

The compilation of results presented in this computational simulation provides evidence that Markov Network representation parameterized with *affinity* factor functions encode sufficient information for solving Raven's matrices of varying difficulty. By solving 44 out of 48 problems, the methodology demonstrates an ability to match different levels of intelligence by strategizing over the Raven's matrix view. By formalizing the concept of compatibility between images, we believe it may be possible to generalize the approach to solving other geometrical problems that encompass mathematical symmetries and more advanced logical progressions.

The most natural generalization of our method is expected for the Advanced Raven Progressive Matrices (APM) test due to their similar structure to SPM. The logical progressions are more complex so an addition of new objective functions may be required to capture the heuristics not observed on the simpler Raven's sets. An extension to Colored Raven Progressive Matrices (CPM) can be achieved by increasing the possible states of the affinity factors from binary (black and white) to arbitrary number of colors with the trade-off of computational speed.

Other geometrical problems, such as Odd One Out, have been addressed within the same family of visual computational models (McGreggor and Goel 2011). While further development is needed to validate the applicability of our approach here, we anticipate a minimal changes in the prediction algorithm (optimizing for less compatible images), and a more substantial effort to infer a different set of rules to explain the patterns in the underlying problems.

Acknowledgments

We thank Drs. Keith McGreggor and Maithilee Kunda for their valuable advices on various stages of this project. Their previous research work has inspired many ideas that spawned the foundations of our structural affinity method. Additionally we would like to thank members of the Design & Intelligence Laboratory, especially David Joyner and Tesca Fitzgerald, for their helpful feedback.

References

Bringsjord, S. 2011. Psychometric artificial intelligence. *Journal of Experimental & Theoretical Artificial Intelligence* 23(3):271–277.

Carpenter, P. A.; Just, M. A.; and Shell, P. 1990. What one intelligence test measures: a theoretical account of the processing in the raven progressive matrices test. *Psychological review* 97(3):404.

Cirillo, S., and Ström, V. 2010. An anthropomorphic solver for raven's progressive matrices (no. 2010: 096). *Goteborg, Sweden: Chalmers University of Technology*.

Evans, T. G. 1964. A heuristic program to solve geometric-analogy problems. In *Proceedings of the April 21-23, 1964, spring joint computer conference*, 327–338. ACM.

Georgiev, N. 2008. Item analysis of c, d and e series from raven's standard progressive matrices with item response theory two-parameter logistic model. *Europe's Journal of Psychology* 4(3).

Hernández-Orallo, J.; Martínez-Plumed, F.; Schmid, U.; Siebers, M.; and Dowe, D. L. 2016. Computer models solving intelligence test problems: Progress and implications. *Artificial Intelligence* 230:74–107.

Hunt, E. 1974. Quote the raven? nevermore. in l. w. gregg (ed.), knowledge and cognition.

Koller, D., and Friedman, N. 2009. *Probabilistic graphical models: principles and techniques*. MIT press.

Kunda, M.; Soulières, I.; Rozga, A.; and Goel, A. K. 2016. Error patterns on the raven's standard progressive matrices test. *Intelligence* 59:181–198.

Kunda, M.; McGreggor, K.; and Goel, A. K. 2013. A computational model for solving problems from the ravens progressive matrices intelligence test using iconic visual representations. *Cognitive Systems Research* 22:47–66.

Kunda, M. 2013. *Visual problem solving in autism, psychometrics, and AI: the case of the Raven's Progressive Matrices intelligence test*. Ph.D. Dissertation, Georgia Institute of Technology.

Little, D. R.; Lewandowsky, S.; and Griffiths, T. 2012. A bayesian model of rule induction in ravens progressive matrices. In *Proceedings of the 34th annual conference of the cognitive science society*, 1918–1923.

Lovett, A.; Tomai, E.; Forbus, K.; and Usher, J. 2009. Solving geometric analogy problems through two-stage analogical mapping. *Cognitive science* 33(7):1192–1231.

Lovett, A.; Forbus, K.; and Usher, J. 2010. A structure-mapping model of raven's progressive matrices. In *Proceedings of the Cognitive Science Society*, volume 32.

Lovett, A.; Lockwood, K.; and Forbus, K. 2008. A computational model of the visual oddity task. In *the Proceedings of the 30th Annual Conference of the Cognitive Science Society. Washington, DC*, volume 25, 29.

McGreggor, K., and Goel, A. 2011. Finding the odd one out: a fractal analogical approach. In *Proceedings of the 8th ACM conference on Creativity and cognition*, 289–298. ACM.

McGreggor, K., and Goel, A. K. 2014. Confident reasoning on raven's progressive matrices tests. In *AAAI*, 380–386.

McGreggor, K.; Kunda, M.; and Goel, A. 2014. Fractals and ravens. *Artificial Intelligence* 215:1–23.

Newell, A. 1983. The heuristic of george polya and its relation to artificial intelligence. *Methods of heuristics* 195–243.

Polya, G. 1945. How to solve it: A new aspect of mathematical model.

Ragni, M., and Neubert, S. 2014. Analyzing ravens intelligence test: Cognitive model, demand, and complexity. In *Computational Approaches to Analogical Reasoning: Current Trends*. Springer. 351–370.

Strannegård, C.; Cirillo, S.; and Ström, V. 2013. An anthropomorphic method for progressive matrix problems. *Cognitive Systems Research* 22:35–46.