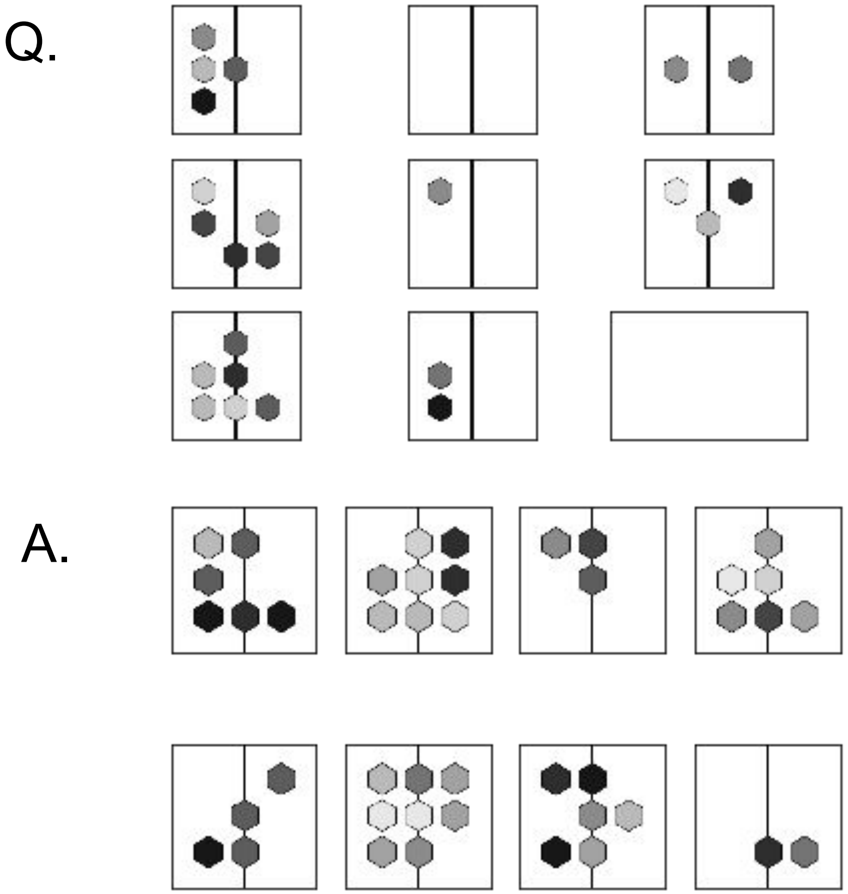
Abstract visual reasoning (problem discussion)

Shashank Shekhar

Masters in Applied Science candidate,
Machine Learning Research Group, University of Guelph
Vector Scholar, Vector Institute

VECTOR INSTITUTE | UNIVERSITY of GUELPH

Raven's Progressive Matrices



- Test of visual intelligence
- Reason about perceptually obvious visual features
- Choose image which completes the matrix
- In cognitive science experiments:

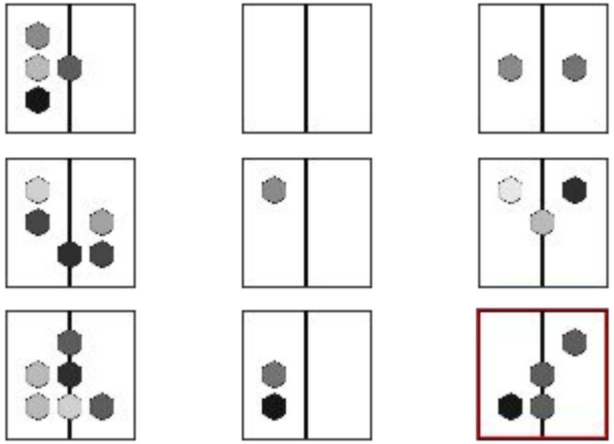
  "RPMs are strongly diagnostic of abstract verbal, spatial and mathematical reasoning ability, discriminating even among populations of highly educated subjects"

VECTOR INSTITUTE | UNIVERSITY of GUELPH

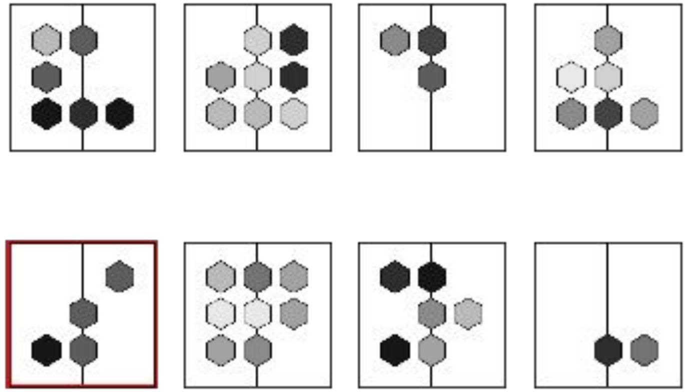
Barrett, David, et al. "Measuring abstract reasoning in neural networks." International Conference on Machine Learning. 2018.

Solution

Q.



A.



Relation structure:

- Type of relation (R) : Progression
- Object of relation (O): Shape
- Attribute of relation (A): Number

i.e. the relation is a progression in the number of shapes (going down the rows of the matrix)

$\{[r,o,a]: r \in R, o \in O, a \in A\}$

Each matrix has 1-4 such relations

Primitives for building abstract features

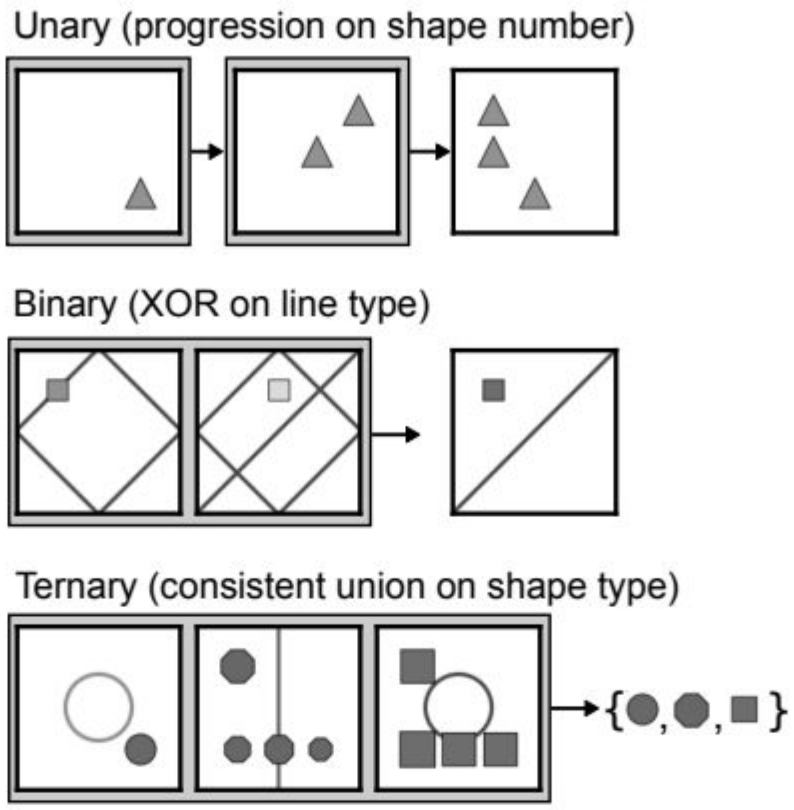
Relation Types (R): Progression, XOR, OR, AND, Consistent Union

Object (O)	Attributes (A)	Possible values (v)
Shape	Size	10 scaling factors evenly spaced in [0, 1]
	Color	10 evenly spaced grayscale intensities in [0, 1]
	Number	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
	Position	((x, y) coordinates in a (0, 1) plot)
	Type	circle, triangle, square, pentagon, hexagon, octagon, star
Line	Type	diagonal down, diagonal up, vertical, horizontal, diamond, circle
	Color	10 evenly spaced grayscale intensities in [0, 1]

Generation process:
(1) Sampling 1- 4 triples (r, o, a)
(2) Sampling values v ∈ V for each a ∈ Sa, adhering to the associated relation r
(3) Sampling values v ∈ V for each a ∉ Sa, ensuring no spurious relation is induced
(4) Rendering the symbolic form into pixels

VECTOR INSTITUTE | UNIVERSITY of GUELPH

Relation types



Categorisation of relation types:

- UNARY (only consider one panel for context)

  e.g. PROGRESSION (P) on the NUMBER (A) of SHAPE (O)

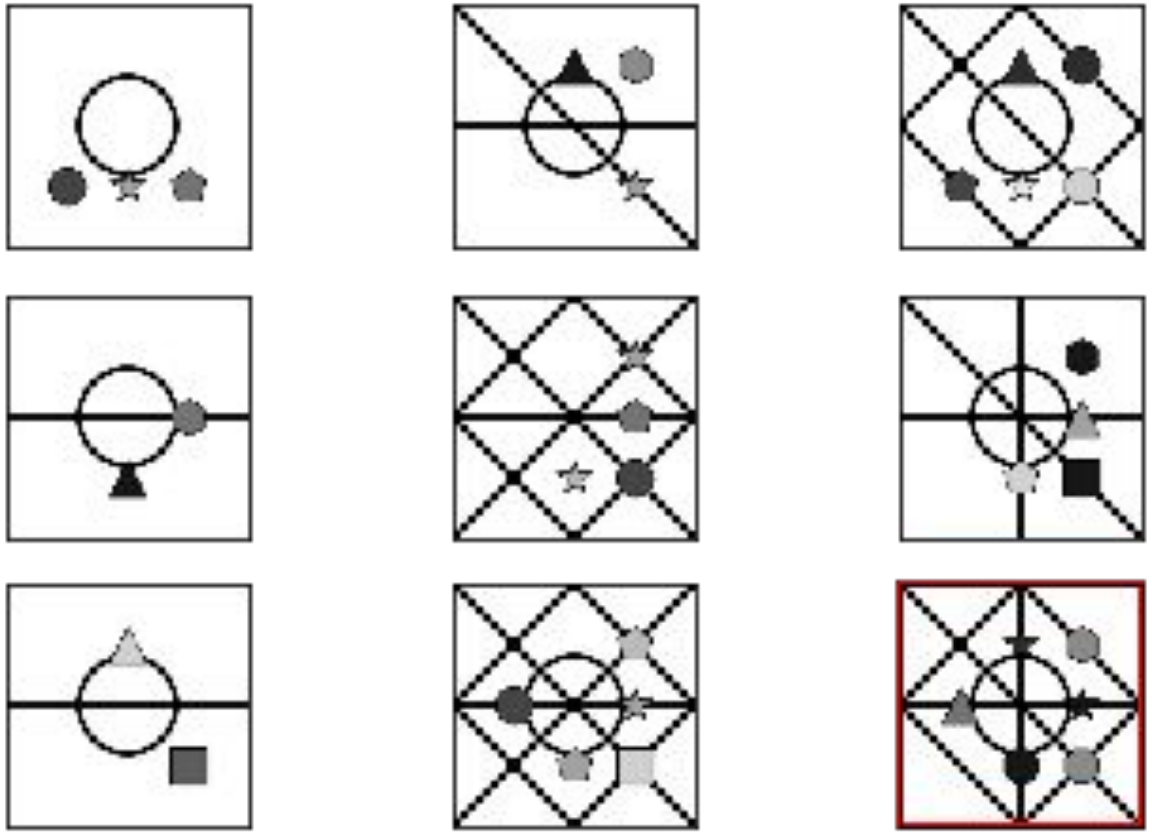
- BINARY (two panels are considered in conjunction to produce third)

  e.g. XOR (R) on the TYPE (A) of LINE (O)

- TERNARY (all three panels adhere to some rule - regardless of order)

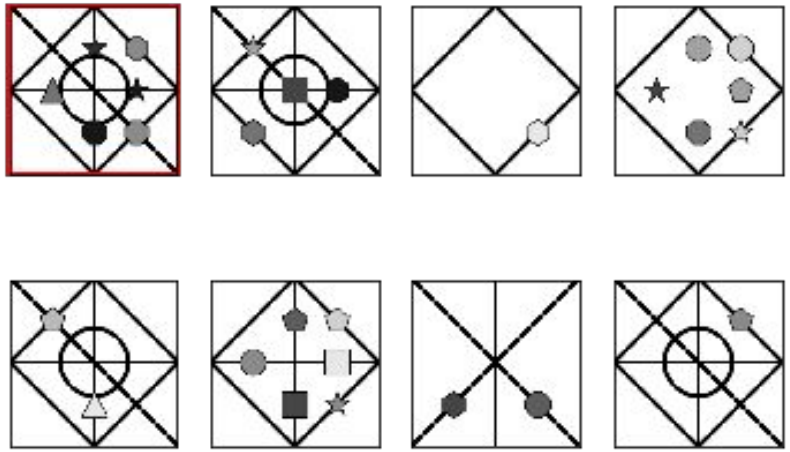
  e.g. CONSISTENT UNION (R) on TYPE (A) of SHAPE (O) i.e. all shapes from common set {circle, hexagon, square} in the example

A hard(er) RPM



Possible relations:

1. OR (R) on POSITION (A) of SHAPES (O) in a row
2. OR (R) on TYPE (A) of LINE (O) in a column



VECTOR INSTITUTE | UNIVERSITY of GUELPH

RPM Generalisation regimes

- Neutral

- Interpolation
- Extrapolation

- Held-out attribute (shape-color)
- Held-out attribute (line-type)

- Held-out triple {r,o,a}
- Held-out Pairs of Triples

- Held-out Attribute Pairs

Neutral

All {r,o,a} triplets that are seen in training are seen in test

-> Difference is just in the pixel-level manifestation of the matrix

Interpolation/Extrapolation

For ordered attributes:

- colour takes 10 evenly spaced grayscale values between [0,1]
- size takes 10 evenly spaced scaling factors between [0,1]
- number takes values 0,1,2,3,4,5,6,7,8,9

Interpolation	**Extrapolation**
train numbers 0,2,4,6,8 test numbers 1,3,5,7,9 (similarly for other attributes)	train numbers 0,1,2,3,4 test numbers 5,6,7,8,9 (similarly for other attributes)

Held-out attribute

Training set S does not contain any triplet with

- o = shape, a = colour (shape-colour)
- o = line, a = type (line-type)

At least one triplet with these held-out attributes is present in test set

Held-out triples/ pair of triples

29 unique triples {r,o,a} in dataset:

- 7 triples in test set (such that each ‘a’ occurs only once, every PGM in the test set has at least one of these triples)

400 viable pairs of triples $(\{r_1,o_1,a_1\},\{r_2,o_2,a_2\}) = (t_1,t_2)$

- 360 train, 40 test
- Any of the 40 (t_1,t_2) do not occur together in train PGM, test PGM has at least one pair out of 40

Held-out pair of attributes

20 (unordered) unique attribute pairs (a_1,a_2) in dataset:

- Such that $(\{r_1,o_1,a_1\},\{r_2,o_2,a_2\})$ is a viable triplet pair
- 16 train, 4 test

RAVEN

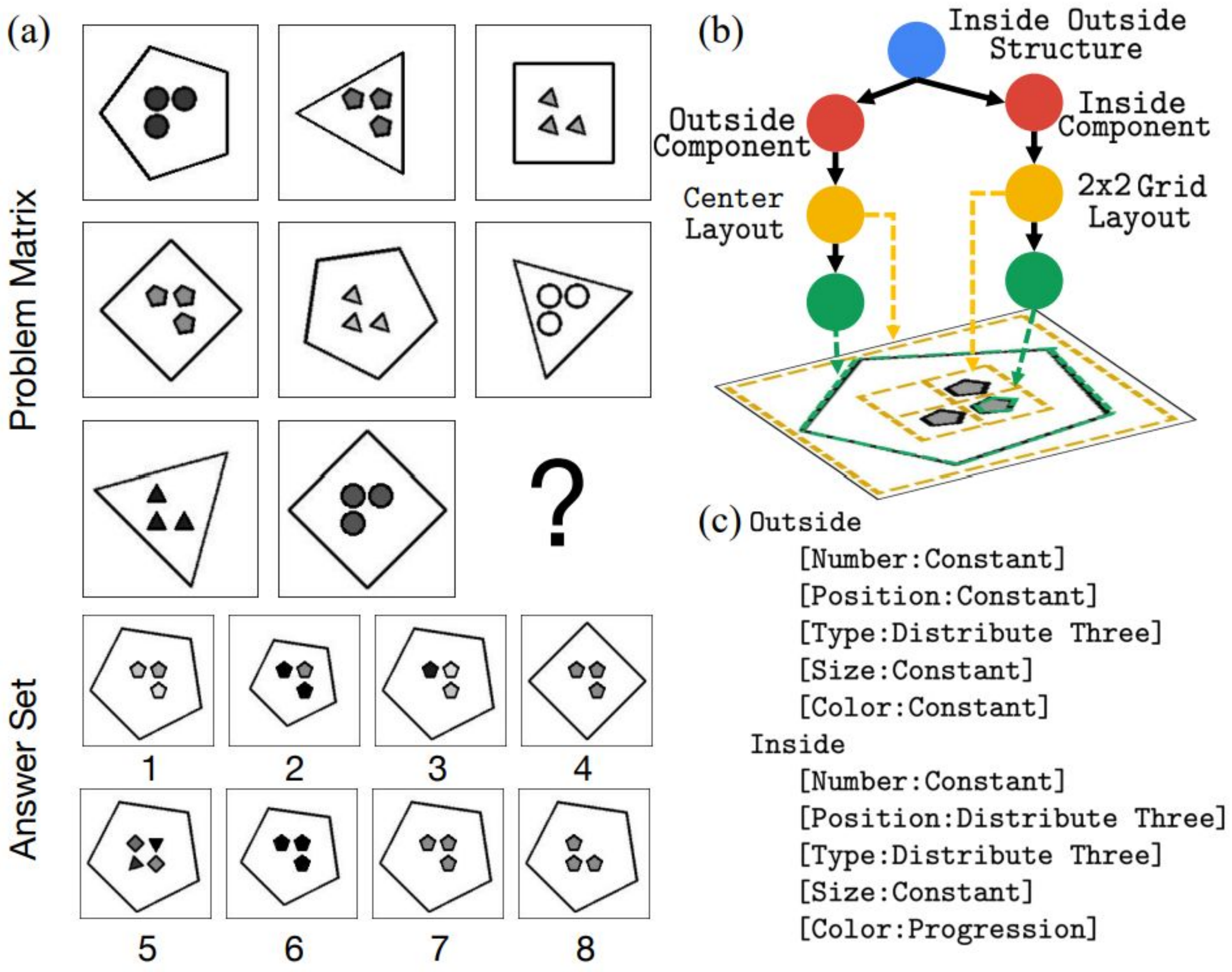
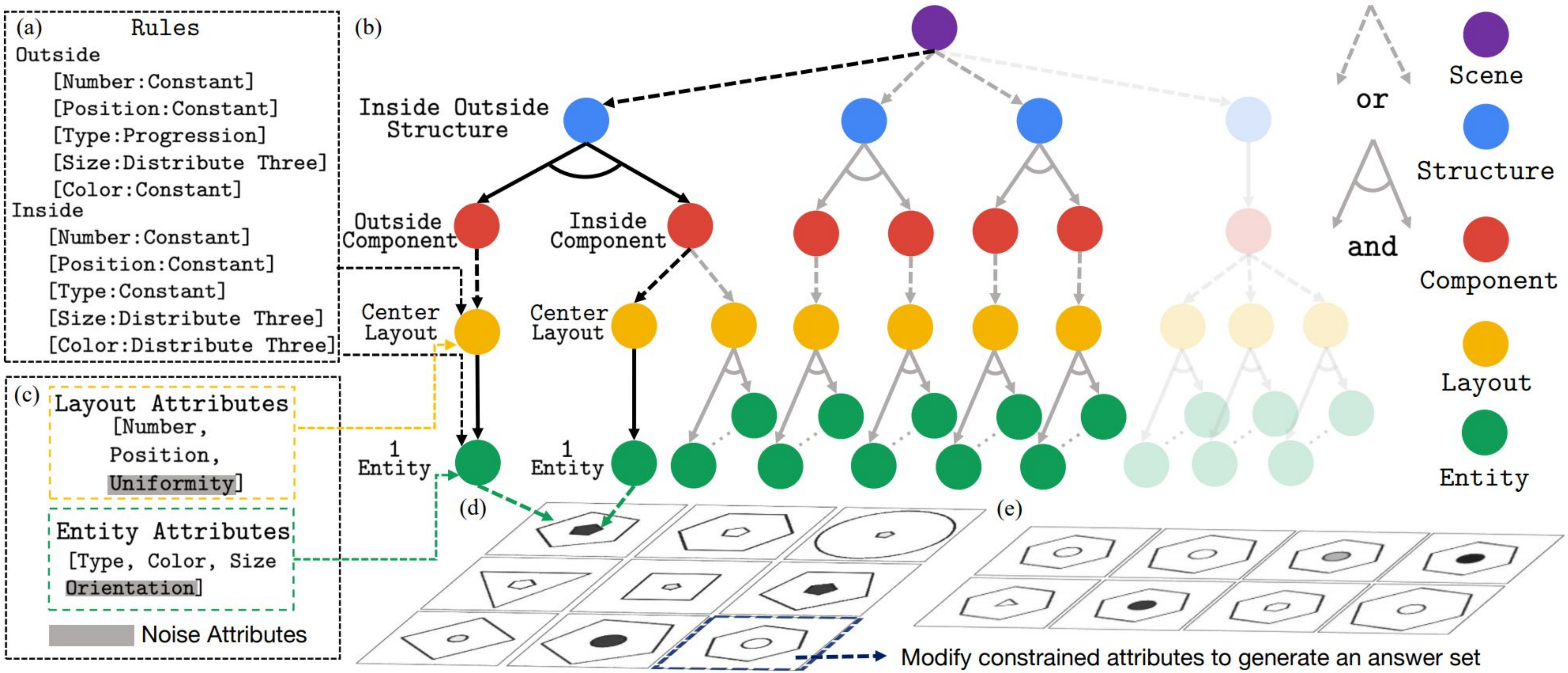


Figure 1. (a) An example RPM. One is asked to select an image that best completes the problem matrix, following the structural and analogical relations. Each image has an underlying structure. (b) Specifically in this problem, it is an inside-outside **structure** in which the outside **component** is a **layout** with a single centered object and the inside **component** is a 2×2 grid **layout**. Details in Figure 2. (c) lists the rules for (a). The compositional nature of the rules makes this problem a difficult one. The correct answer is 7.

RAVEN	PGM
More structure + structured annotations (generated using Attributed Stochastic Image Grammar)	Less structure
More rules per RPM	Less rules
7 figure configs	3 fig configs
Fewer* samples (*70k)	1.2M train set

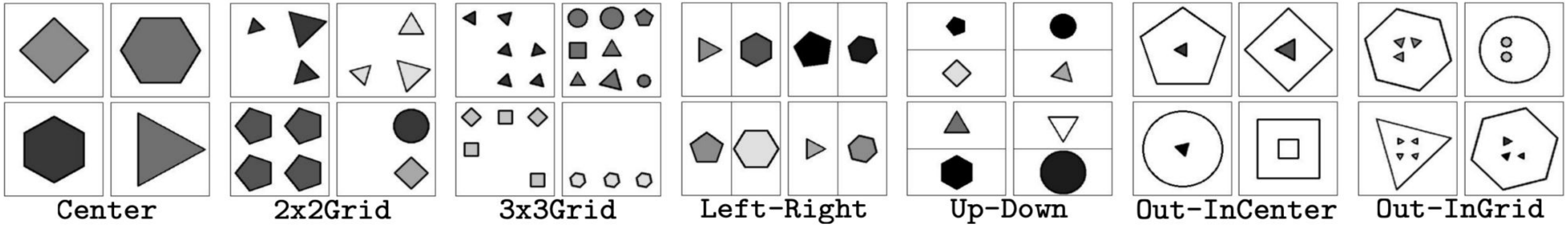
Zhang, C., Gao, F., Jia, B., Zhu, Y., & Zhu, S. C. (2019). Raven: A dataset for relational and analogical visual reasoning. CVPR 2019

RAVEN



(a) Rules
Outside
[Number:Constant]
[Position:Constant]
[Type:Progression]
[Size:Distribute Three]
[Color:Constant]
Inside
[Number:Constant]
[Position:Constant]
[Type:Constant]
[Size:Distribute Three]
[Color:Distribute Three]
(b)
Inside Outside Structure
Outside Component
Inside Component
Center Layout
Center Layout
1 Entity
1 Entity
or
and
Scene
Structure
Component
Layout
Entity
(c) Layout Attributes [Number, Position, Uniformity]
Entity Attributes [Type, Color, Size Orientation]
Noise Attributes
(d)
(e)
Modify constrained attributes to generate an answer set

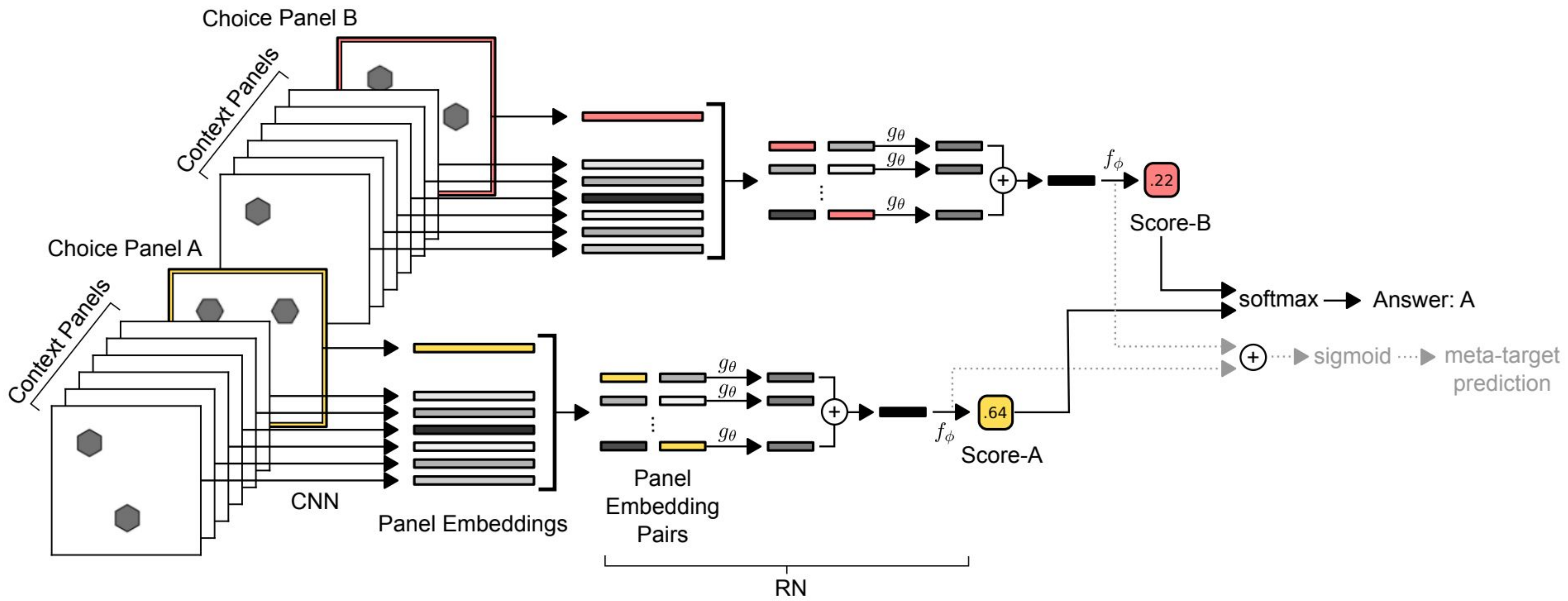
Generalisation regimes



Regime	Measure
Train: Center **Test:** Left-Right, Up-Down, and Out-InCenter	"compositional reasoning ability of the model as it requires the model to generalize the rules learned in a single-component configuration to configurations with multiple independent but similar components"
Train: Left-Right **Test:** Up-Down (and vice versa)	"...one could be regarded as a transpose of another. Thus, the test could measure whether the model simply memorizes the pattern in one configuration."
Train: 2x2 Grid **Test:** 3x3 Grid (and vice versa)	"Both configurations involve multi-object interactions. Therefore, the test could measure the generalization when the number of objects changes"

Solution approaches

Wild Relation Network



f,g are MLPs (constitute the relation network) - look at pairs of panel embeddings to extract relations - and then combine them across all pairs

VECTOR INSTITUTE | UNIVERSITY of GUELPH

Results (PGM)

Model	Test (%)
WReN	**62.6**
Wild-ResNet	48.0
ResNet-50	42.0
LSTM	35.8
CNN + MLP	33.0
Blind ResNet	22.4

	$\beta = 0$			$\beta = 10$		
Regime	**Val. (%)**	**Test (%)**	**Diff.**	**Val. (%)**	**Test (%)**	**Diff.**
Neutral	63.0	62.6	-0.6	77.2	76.9	-0.3
Interpolation	79.0	64.4	-14.6	92.3	67.4	-24.9
H.O. Attribute Pairs	46.7	27.2	-19.5	73.4	51.7	-21.7
H.O. Triple Pairs	63.9	41.9	-22.0	74.5	56.3	-18.2
H.O. Triples	63.4	19.0	-44.4	80.0	20.1	-59.9
H.O. `line-type`	59.5	14.4	-45.1	78.1	16.4	-61.7
H.O. `shape-colour`	59.1	12.5	-46.6	85.2	13.0	-72.2
Extrapolation	69.3	17.2	-52.1	93.6	15.5	-78.1

Almost no better than random (12.5%) !!

Other observations (PGM)

- “worse generalisation in the case of Held-out Triples suggests that the model was **less able to induce the meaning** of unfamiliar triples **from its knowledge of their constituent components**”

- More relations in PGM - poorer performance

- “the pressure to represent abstract semantic principles such that they can be decoded simply into **discrete symbolic explanations** seems to improve the ability of the model to **productively compose its knowledge**”

- “.....,for the relation property, the difference between a correct and incorrect meta-target prediction was substantial (86.8% vs. 32.1%). This result suggests that **predicting the relation property correctly is most critical to task success**” (for WReNs)

VECTOR INSTITUTE | UNIVERSITY of GUELPH

Dynamic Residual Tree

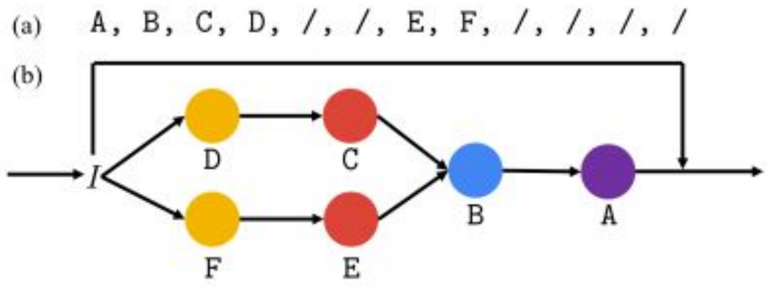


Figure 5. An example computation graph of DRT. (a) Given the serialized n-ary tree representation (pre-order traversal with / denoting end-of-branch), (b) a tree-structured computation graph is dynamically built. The input features are wired from bottom-up following the tree structure. The final output is the sum with the input, forming a residual module.

Using the pre-order traversal of the A-SIG, a tree structure is built (each node represents Layout, Component, Structure, Scene etc)

Each node is a fully connected layer (instead of LSTM cell in Tree-LSTM) updated as :

$$I = \mathrm{ReLU}\left(f\left(\left[\sum_c I_c, w_n\right]\right)\right)$$

w are word vector representations of the node label, Ic are input features from child node

The bottom level input I is just features from a CNN

DRT results (RAVEN)

Method	Acc	Center	2x2Grid	3x3Grid	L-R	U-D	O-IC	O-IG
LSTM	13.07%	13.19%	14.13%	13.69%	12.84%	12.35%	12.15%	12.99%
WReN	14.69%	13.09%	28.62%	28.27%	7.49%	6.34%	8.38%	10.56%
CNN	36.97%	33.58%	30.30%	33.53%	39.43%	41.26%	43.20%	37.54%
ResNet	53.43%	52.82%	41.86%	44.29%	58.77%	60.16%	63.19%	53.12%
LSTM+DRT	13.96%	14.29%	15.08%	14.09%	13.79%	13.24%	13.99%	13.29%
WReN+DRT	15.02%	15.38%	23.26%	29.51%	6.99%	8.43%	8.93%	12.35%
CNN+DRT	39.42%	37.30%	30.06%	34.57%	45.49%	45.54%	45.93%	37.54%
ResNet+DRT	**59.56%**	**58.08%**	**46.53%**	**50.40%**	**65.82%**	**67.11%**	**69.09%**	**60.11%**
Human	84.41%	95.45%	81.82%	79.55%	86.36%	81.81%	86.36%	81.81%
Solver*	100%	100%	100%	100%	100%	100%	100%	100%

Generalisation Results (RAVEN)

Table 3. Generalization test. The model is trained on `Center` and tested on three other configurations.

Center	Left-Right	Up-Down	Out-InCenter
51.87%	40.03%	35.46%	38.84%

Table 4. Generalization test. The row shows configurations the model is trained on and the column the model is tested on.

	Left-Right	Up-Down
Left-Right	41.07%	38.10%
Up-Down	39.48%	43.60%

Table 5. Generalization test. The row shows configurations the model is trained on and the column the model is tested on.

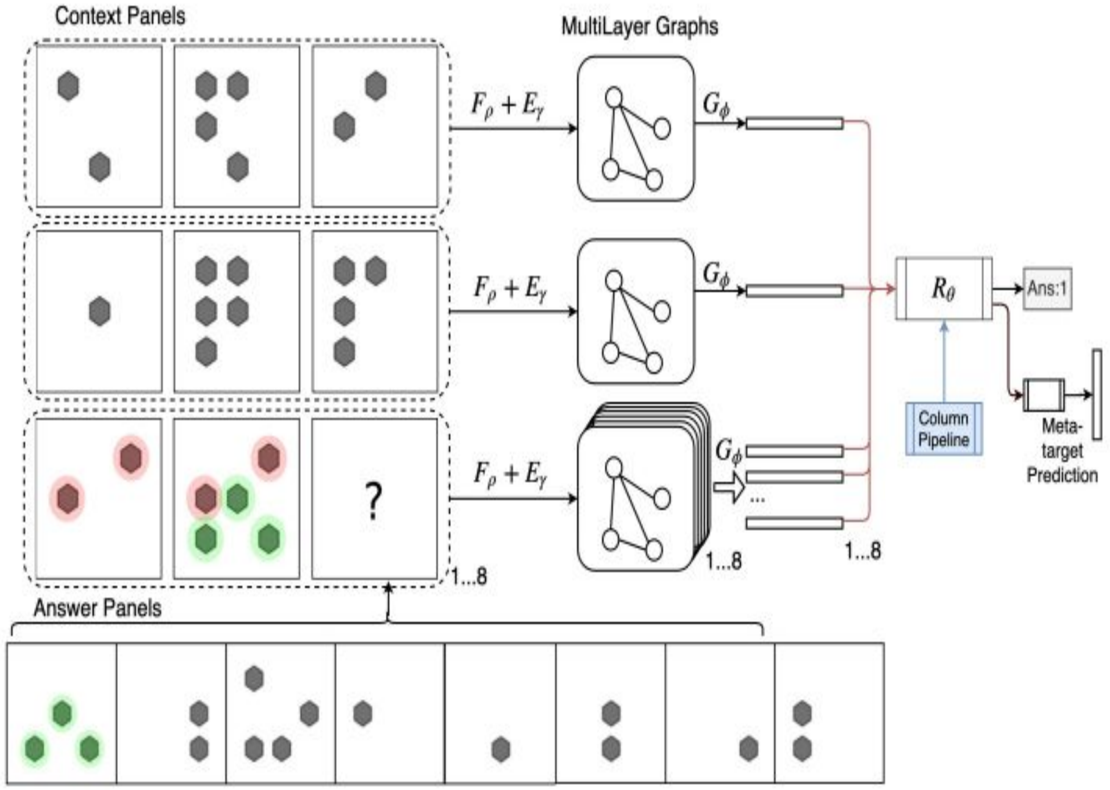
	2x2Grid	3x3Grid
2x2Grid	40.93%	38.69%
3x3Grid	39.14%	43.72%

Other observations (RAVEN)

- "WReN achieves higher accuracy on configurations consisting of multiple randomly distributed objects (2x2Grid and 3x3Grid), **with drastically degrading performance in configurations consisting of independent image components**. This suggests WReN is biased to grid-like configurations (majority of PGM) but not others that require compositional reasoning (as in RAVEN)"

- Both ResNet+DRT and WReN+DRT suffer performance loss on meta-target prediction and structured annotation prediction (exactly opposite to PGM observations!)

Multiplex Multilayer Graph Net (MXGNet)



(a)

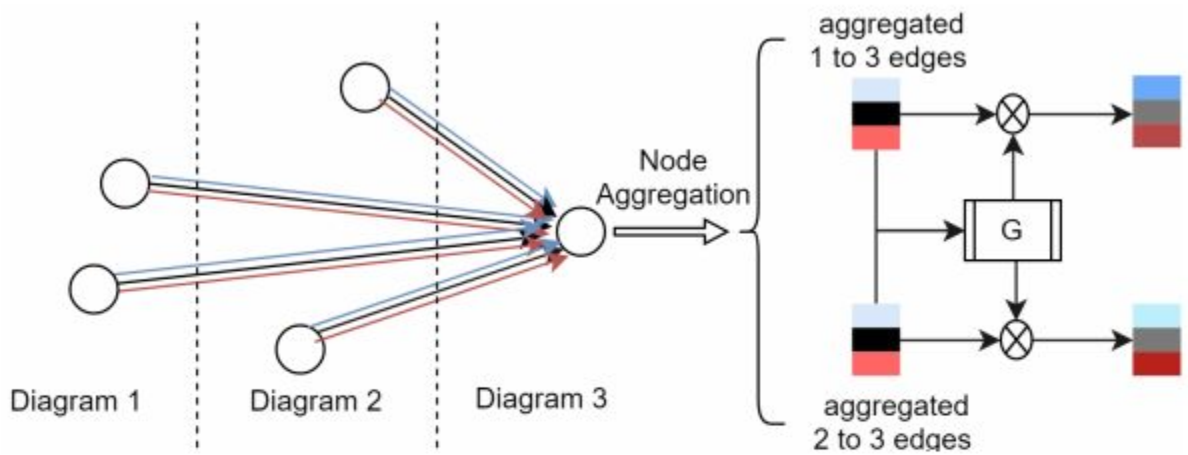
- **Fp, object-representation module:** CNN: each location is treated as object feature vector in grid features

  Spatial attention: Attend location of object -> extract using CNN; for each location z_{pres} indicates whether object present

- **E_y, Edge embeddings module**
- **G_phi, Graph summarization module**
- **R_theta, Reasoning network**

Wang, Duo, Mateja Jamnik, and Pietro Lio. "Abstract diagrammatic reasoning with multiplex graph networks." ICLR 2020.

MXGNet: (shoddy) explanation



Multiplex edge embeddings:

Fg returns object representations v_{ij}
$i \subset [1,N]$ #frames (use only row/col)
$j \subset [1,L]$ #objects (nodes in a layer)

$$e^t_{(i,j),(l,k)} = E^t_\gamma(P^k(v_{i,j}, v_{l,k}))$$

P = Projection layer projecting concatentated node embeddings to T different embeddings

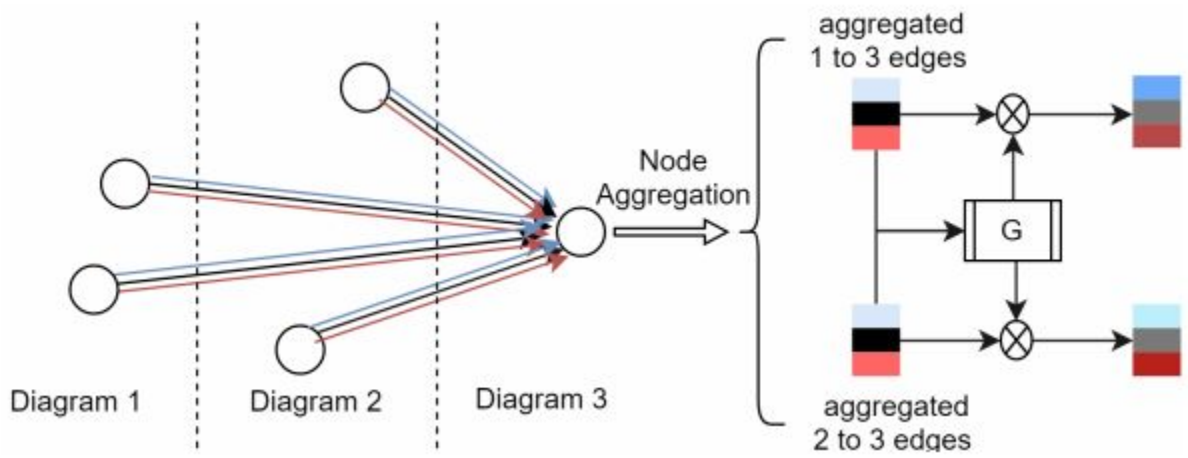
E = MLP processing t^{th} projections to t^{th} layer of edge embeddings

Graph summarization:

Concat the max(), min(), sum(), mean() of all edges from nodes in a particular layer to nodes in last layer (since relations are of the form:

Diagram 3 = F(Diagram 1, Diagram 2))

MXGNet: (shoddy) explanation



Reasoning network: Takes relational embeddings and ranks all candidate answers using ResNet + softmax

Cross-multiplexing gating:

Takes aggregated node info from each layer -> outputs gating variable for each node in layer (implemented as multi-head MLP)

Finally, take node embeddings, multiply with gating function, pass through MLP = node embeddings

Take all node embeddings, concatenate, pass through ResBlock = Relation feature embedding for subset (e.g. row 1/2/3)

MXGNet: results

Model	WReN Barrett et al. (2018)	VAE-WReN Steenbrugge et al. (2018)	ARNe Anonymous (2020)	MXGNet CNN	MXGNet Sp-Attn
acc. (%) $\beta = 10$	76.9	N/A	88.2	**89.6**	88.8
acc. (%) $\beta = 0$	62.6	64.2	N/A	**66.7**	66.1

(a) PGM

Model	WReN Zhang et al. (2019)	ResNet Zhang et al. (2019)	ResNet+DRT Zhang et al. (2019)	ARNe Anonymous (2020)	MXGNet CNN	MXGNet Sp-Attn
acc. (%)	14.69	53.43	59.56	19.67	**83.91**	82.61

(b) RAVEN

MXGNet: results

Model	Regime	$\beta = 0$ Val.(%)	$\beta = 0$ test%	$\beta = 0$ Diff.	$\beta = 10$ Val.(%)	$\beta = 10$ test%	$\beta = 10$ Diff.
WReN	Neutral	63.0	62.6	-0.4	77.2	76.9	-0.3
	Interpolation	79.0	64.4	-14.6	92.3	67.4	-24.9
	Extrapolation	69.3	17.2	-52.1	93.6	15.5	-79.1
MXGNet	Neutral	67.1	**66.7**	-0.4	89.9	**89.6**	-0.3
	Interpolation	74.2	**65.4**	-8.8	91.5	**84.6**	-6.9
	Extrapolation	69.1	**18.9**	-50.2	94.3	**18.4**	-75.9

VECTOR INSTITUTE | UNIVERSITY of GUELPH

Evaluation flaw in RAVEN

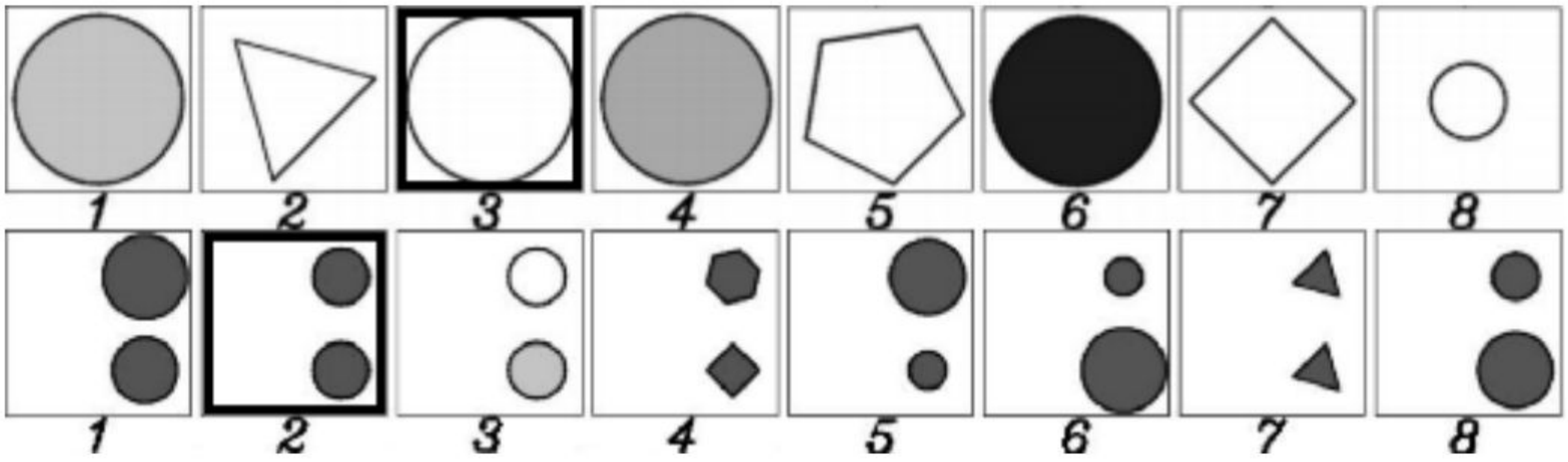


Fig. 2. Two example answer sets from problems in RAVEN. We can derive the correct answer (emboldened) from each set by finding the intersection of the set's modes of shape, colour, and scale factors. Essentially, "which frame has the most common features?"

Steven Spratley, Krista Ehinger, and Tim Miller. A Closer Look at Generalisation in RAVEN. ECCV 2020

Evaluation flaw in RAVEN

Table 1. Accuracy (%) of ResNet and Rel-Base, trained context-blind on RAVEN.

	Acc	Centre	2x2	3x3	L-R	U-D	O-IC	O-IG
ResNet	83.11	84.23	65.34	68.70	95.14	95.82	92.02	80.53
Rel-Base	92.46	98.49	78.66	80.52	99.22	99.66	98.63	92.04

Steven Spratley, Krista Ehinger, and Tim Miller. A Closer Look at Generalisation in RAVEN. ECCV 2020

Role of symbolic knowledge & relational bias

Previous methods relied (heavily) on using meta targets as well as strong inductive biases for learning relations.

Are they necessarily needed?

Can we distengale the underlying objects (factors) and simply pass to neural network?

In PGMs

Xander Steenbrugge, Sam Leroux, Tim Verbelen, and Bart Dhoedt. **Improving generalization for abstract reasoning tasks using disentangled feature representations.** Neural Information Processing Systems (NeurIPS) Workshop on Relational Representation Learning, 2018

Replaces CNN with VAE in the original WReN approach:

Model-type	CNN-WReN [1]			VAE-WReN (β=4.00)		
Generalization regime	Val (%)	Test (%)	Test (kappa)	Val (%)	Test (%)	Test (kappa)
Neutral	63.0	62.6	0.573	**64.8**	**64.2**	**0.591**
H.O. Triple Pairs	63.9	41.9	0.336	**64.6**	**43.6**	**0.355**
H.O. Attribute Pairs	46.7	27.2	0.168	**70.1**	**36.8**	**0.278**
H.O. Triples	**63.4**	19.0	0.074	59.5	**24.6**	**0.138**

Showed some-level of object disentanglement in PGM scenes

Question:

NIPS Proceedings$^{\beta}$ Books 2019

Are Disentangled Representations Helpful for Abstract Visual Reasoning?

Part of: Advances in Neural Information Processing Systems 32 (NIPS 2019)

[PDF] [BibTeX] [Supplemental] [Reviews] [Author Feedback] [Meta Review] [Sourcecode]

Authors

- Sjoerd van Steenkiste
- Francesco Locatello
- Jürgen Schmidhuber
- Olivier Bachem

Extensive study of different VAE models with Relation network for visual reasoning tasks (not PGMs, another task where underlying factors were controlled). Conclusions:

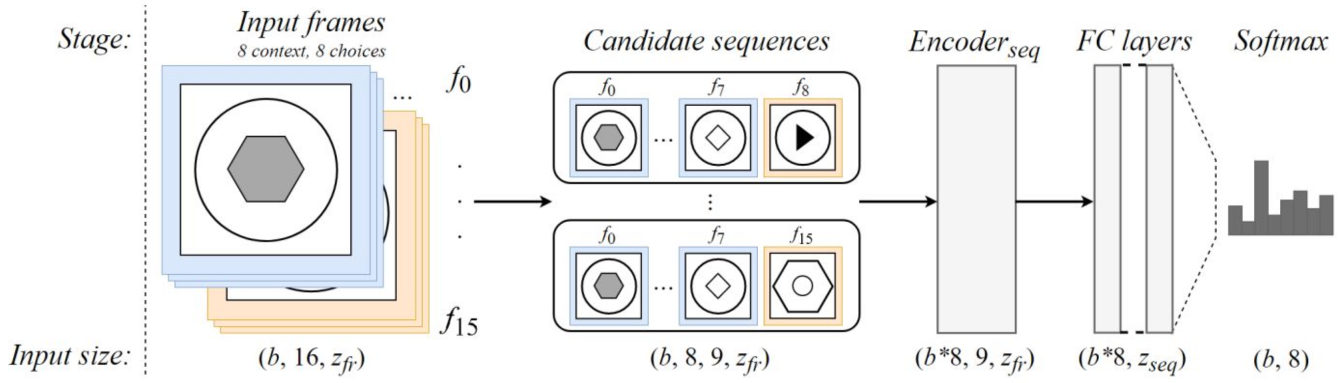
1. “these results provide concrete motivation why one might want to pursue disentanglement as a property of learned representations in the unsupervised case.”

2. “.. observed differences between disentanglement metrics, which should motivate further work in understanding what different properties they capture.”

3. “....useful to extend the methodology in this study to other complex down-stream tasks, or include an investigation of other purported benefits of disentangled representations”

Answer: Yes

Object/Frame Relational ResNet

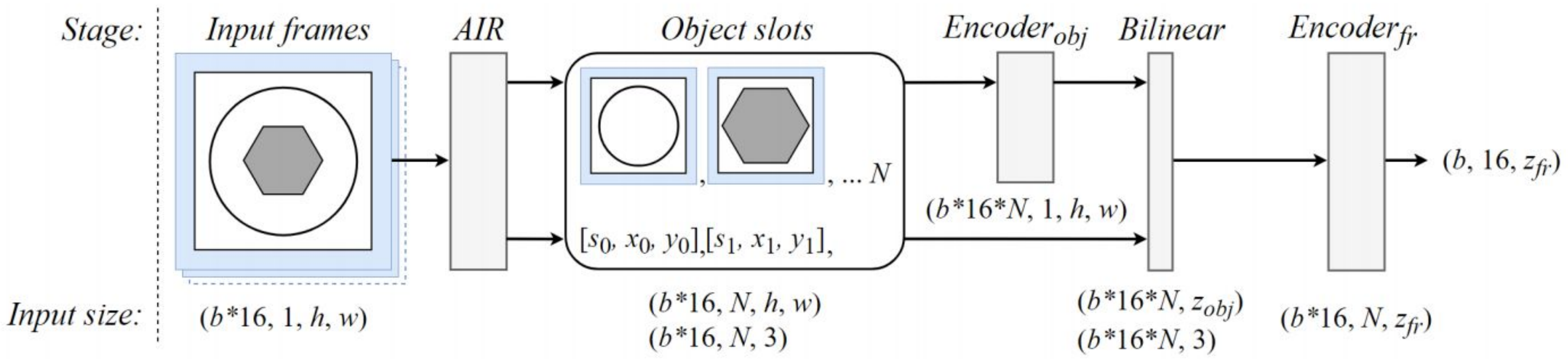


ResNet baseline: Stack frames into independent sequences (one frame per candidate), pass through 4-layer ResNet and then rank using FC layers and softmax.
(*Different from original paper approach as all candidate frames were processed at once)

Frame-relational ResNet (Rel-Base): Two stage-

- Take all frames and embed them individually (using ResNet)
- Take frame embeddings, stack into candidate sequences (as above), and pass through 1D convolution. This enables to learn low-level perceptual processes unaffected by position of frames, and high level that models relations in & between embeddings.

Rel-AIR: explanation



- **Scene decomposition:** Uses attend-infer-repeat (a sort of iterative VAE which splits up the scene into object latents) to get object slots, scales and positions
- **Object embedding:** Encode objects through CNN
- **Latent-informed object embedding:** Combine object embedding with scale, position and pass the paired data through a bilinear layer to unify
- **Object-relational feature extraction:**Reshape the object-embedding into N object channels, pass through 1D residual encoder to generate frame embeddings

Finally, these object-relational feature embeddings are stacked into sequences, encoded and scored using fully-connected layers (same as Rel-Base)

Rel-Base and Rel-AIR: results

PGM set	Wild-ResNet [20]	WReN	CoPINet [29]	LEN	LEN*	LEN**	Rel-Base
Neutral	48.00	62.60	56.37	68.10	70.30	85.10	**85.50**
Extrapolation	N/A	17.20	N/A	N/A	N/A	N/A	**22.05**

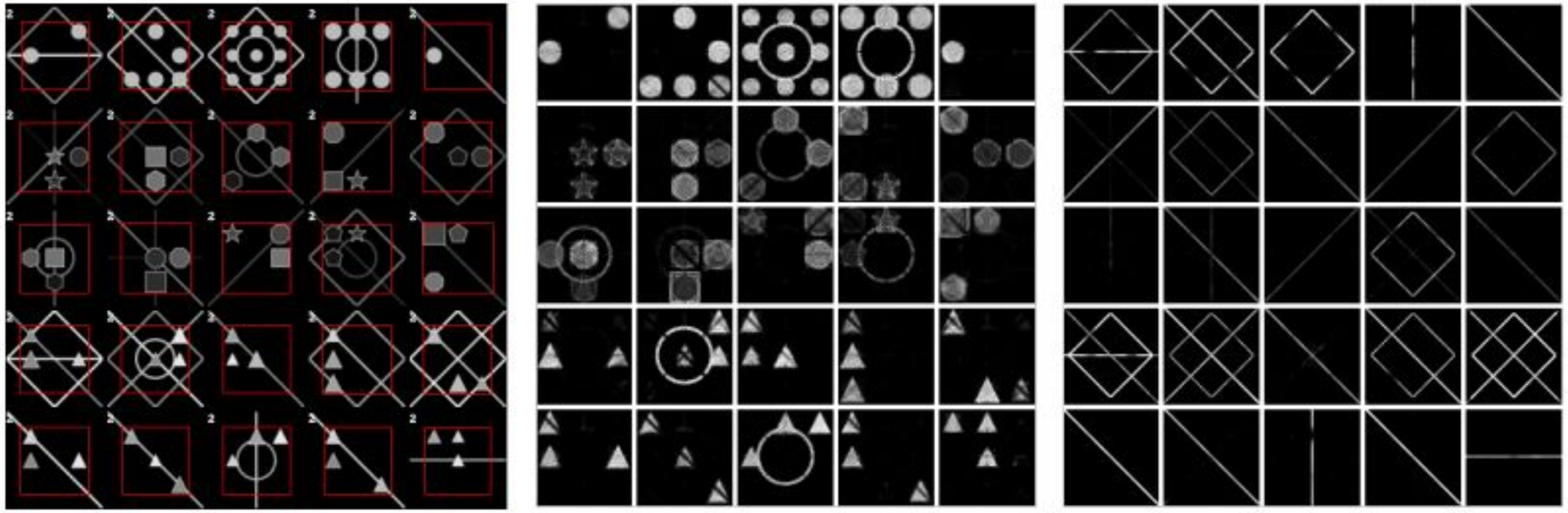


Fig. 6. AIR decomposes PGM frames (left) into grid and background slots (centre, right). Red bounding boxes denote attention windows for the first slot.

Rel-Base and Rel-AIR: results

Method	Acc	Centre	2x2	3x3	L-R	U-D	O-IC	O-IG
WReN [29]	17.9	15.4	29.8	32.9	11.1	11.0	11.1	14.5
ResNet	34.5	41.7	34.1	38.5	33.4	31.7	34.6	27.3
LEN [30]	72.9	80.2	57.5	62.1	73.5	81.2	84.4	71.5
LEN+T [30]	78.3	82.3	58.5	64.3	87.0	85.5	88.9	81.9
Human [28]	84.4	95.5	81.8	79.6	86.4	81.8	86.4	81.8
Rel-Base	91.7	97.6	85.9	86.9	93.5	96.5	97.6	83.8
Rel-AIR	**94.1**	**99.0**	**92.4**	**87.1**	**98.7**	**97.9**	**98.0**	**85.3**

Table 4. Generalisation test between `Left-Right` and `Up-Down` configurations. Rows and columns indicate training and test sets respectively.

	Left-Right			Up-Down		
	ResNet	Rel-Base	Rel-AIR	ResNet	Rel-Base	Rel-AIR
`Left-Right`	27.83	90.09	**98.07**	3.71	32.71	**66.77**
`Up-Down`	2.98	22.61	**60.81**	26.42	90.23	**94.84**

% of training set	ResNet	Rel-Base	Rel-AIR
10	14.79	24.40	**51.39**
25	21.48	52.24	**81.07**
100	34.51	91.66	**94.10**

Table 5. Generalisation test between `2x2Grid` and `3x3Grid` configurations. Rows and columns indicate training and test sets respectively.

	2x2Grid			3x3Grid		
	ResNet	Rel-Base	Rel-AIR	ResNet	Rel-Base	Rel-AIR
`2x2Grid`	26.32	60.16	**88.24**	13.96	41.55	**67.01**
`3x3Grid`	14.36	34.03	**61.90**	33.84	68.16	**82.54**

Availability of object lists reduces problem complexity greatly

Open questions

Open questions: compositional generalization

If a model has seen certain relation in $\{o_1,a_1\}$ and never seen it in $\{o_1,a_2\}$ how well is it able to generalise (i.e. compose the relation for unseen attribute of the same object)

- Similarly hold attribute constant and vary object
- Finally vary both

This can be seen as a better measure of understanding a relation across visual concepts. This can also evaluate ‘object-centric ness’ of object centric representations

- How does this relate to type of relation (unary/binary/ternary)?

Open questions: role of inductive biases

Two different directions of inductive biases:

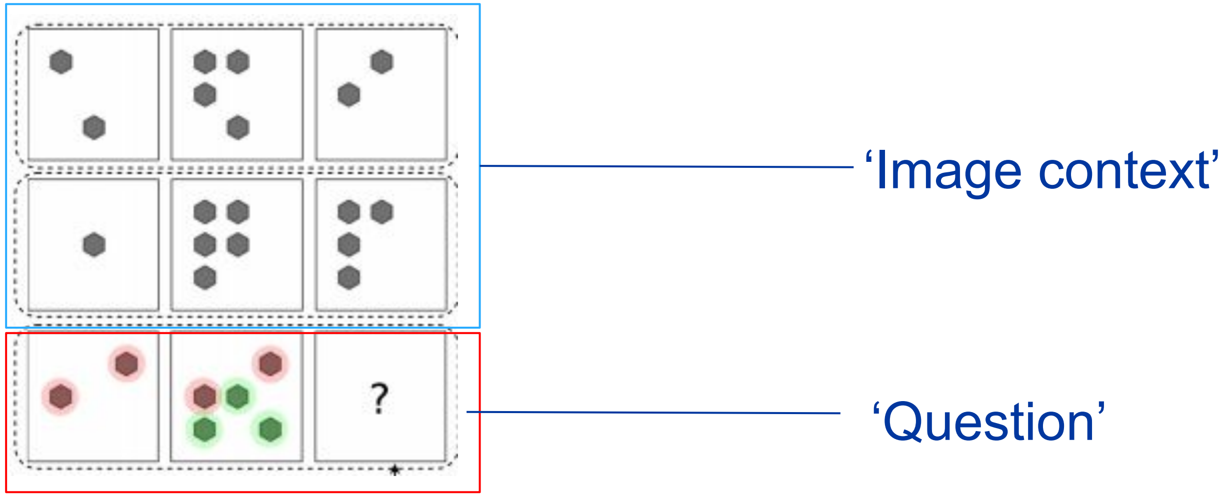
- Object-centric representations (VAE, Rel-AIR)
- Relation learning (WReN, MXGNet)

How does generalization differ across both?

- Do better object-centric representations lead to better generalization across object/attribute types?
- Does strong relation leaning bias reduce possibility of generalizing across unseen relation types?

Open questions: can we adopt methods from VQA?

Can possibly use something similar for abstract reasoning:



Open (closed?) questions: modular networks

Modular networks are used in CLEVR (VQA task with compositional requirements) and other tasks:

- Hu, Ronghang, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, Kate Saenko. **"Learning to reason: End-to-end module networks for visual question answering."** CVPR 2017
- Drew A Hudson and Christopher D Manning. **“Compositional attention networks for machine reasoning”**. ICLR, 2018
- Michael Chang, Abhishek Gupta, Sergey Levine, and Thomas L. Griffiths **“Automatically composing representation transformations as a means for generalization”** ICLR 2019

Also being used* for abstract visual reasoning (NeurIPS 2020 submissions on arxiv)

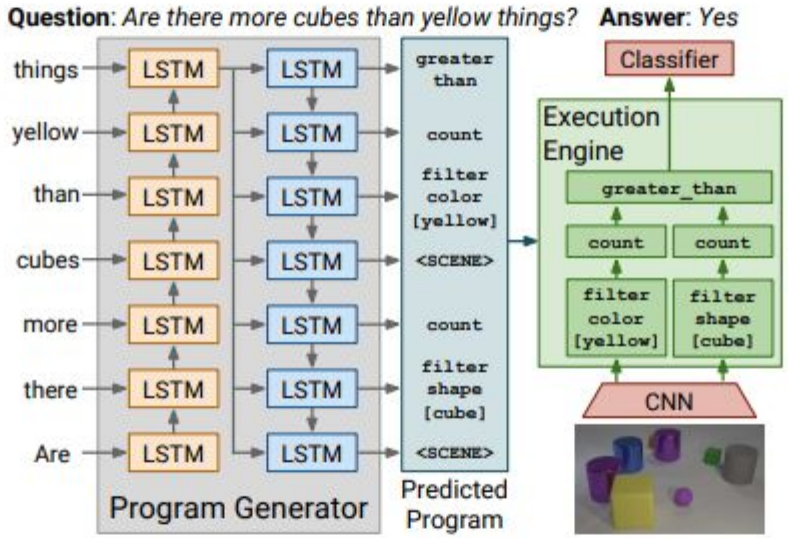
- Yuhuai Wu, Honghua Dong, Roger Grosse, Jimmy Ba. **“The Scattering Compositional Learner: Discovering Objects, Attributes, Relationships in Analogical Reasoning”,** arxiv 2020
- Xiangru Tang, Haoyuan Wang, Xiang Pan, Jiyang Qi, **“Multi-Granularity Modularized Network for Abstract Visual Reasoning”,** arxiv 2020

Does that mean we can use other ideas from VQA?

* yet to review

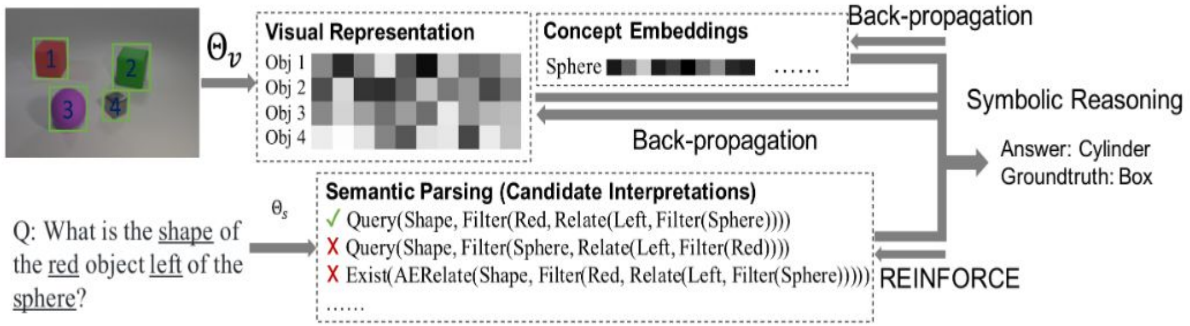
Open questions: Program synthesis approach

Program synthesis approaches have been used in CLEVR. Seeing how PGM and RAVEN are both procedurally generated problems, program induction/synthesis seems like an obvious approach to try.



Neural

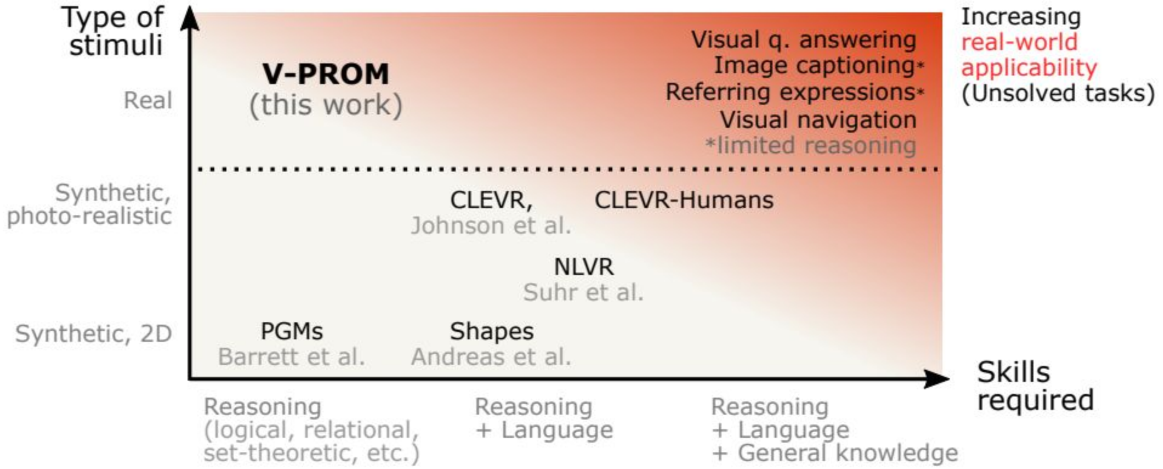
- Johnson, J., Hariharan, B., Van Der Maaten, L., Hoffman, J., Fei-Fei, L., Lawrence Zitnick, C. and Girshick, R., **“Inferring and executing programs for visual reasoning”**. CVPR 2017.



Neural+Symbolic

- **“The Neuro-Symbolic Concept Learner: Interpreting Scenes, Words, and Sentences From Natural Supervision”** Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B. Tenenbaum, and Jiajun Wu. ICLR 2019

Open questions: Scaling to reasoning on real images



VQA approaches can be especially helpful for real-world analogical reasoning problems.......

VECTOR INSTITUTE | UNIVERSITY of GUELPH

Teney, Damien, et al. "V-PROM: A Benchmark for Visual Reasoning Using Visual Progressive Matrices." AAAI. 2020.

Thanks

Credits
Eric for slide layout
Graham for initial problem discussion