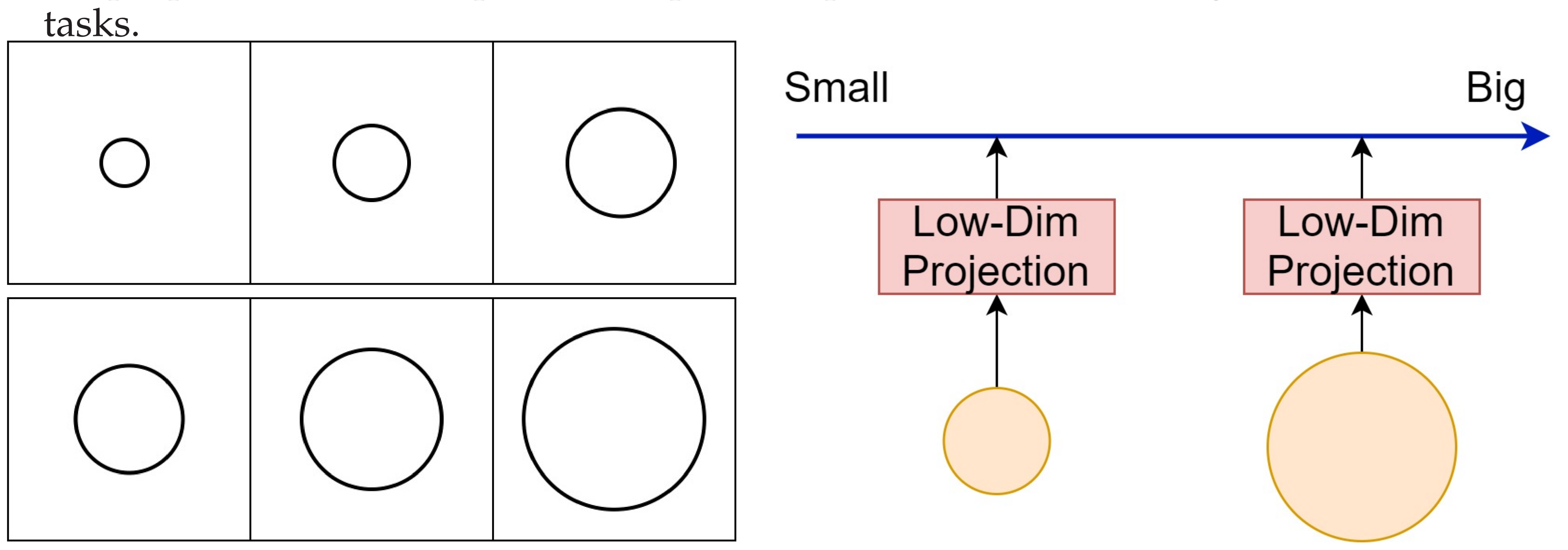
Extrapolatable Relational Reasoning With Comparators in Low-Dimensional Manifolds

Duo Wang[1], Mateja Jamnik[1] and Pietro Lio[1]
[1]University of Cambridge

UNIVERSITY OF CAMBRIDGE

Introduction

- Deep Neural Networks often have poor out-of-distribution (**o.o.d**) generalisation capability, meaning that its performance degrades when test data distribution differ from training distribution. This is particularly the case for relational reasoning tasks.
- Human can **o.o.d** generalise better. For example, In figure 1.a, when human are trained to recognise progression relation in the size of circles for smaller circles, they can generalise to bigger circles without re-training. We hypothesise that this is because human brains have neurons (e.g. LIP neuron) that respond to only certain attributes of objects (e.g. size), thereby creating information bottleneck and improve generalisation.
- This behaviour is equivalent to projecting object representations to low-dimensional space (Figure 1.b). We developed low-dim comparator module based on this idea, and shown that the proposed module improve extrapolation performance for a range of relational reasoning tasks.



(a) (b)

1.(a) Circle size progression (b) Low-Dim Projection.

Comparator in Low-dimensional Manifolds

The inductive bias module is comprised of low-dim projection functions p and comparators c. Let $\{o_i; i \in 1 \ldots N\}$ be the set of object representations, obtained by extracting features from raw inputs such as applying Convolutional Neural Networks (CNN) on images. Pairwise comparison between object pair o_i and o_j can be achieved with a function f expressed as:

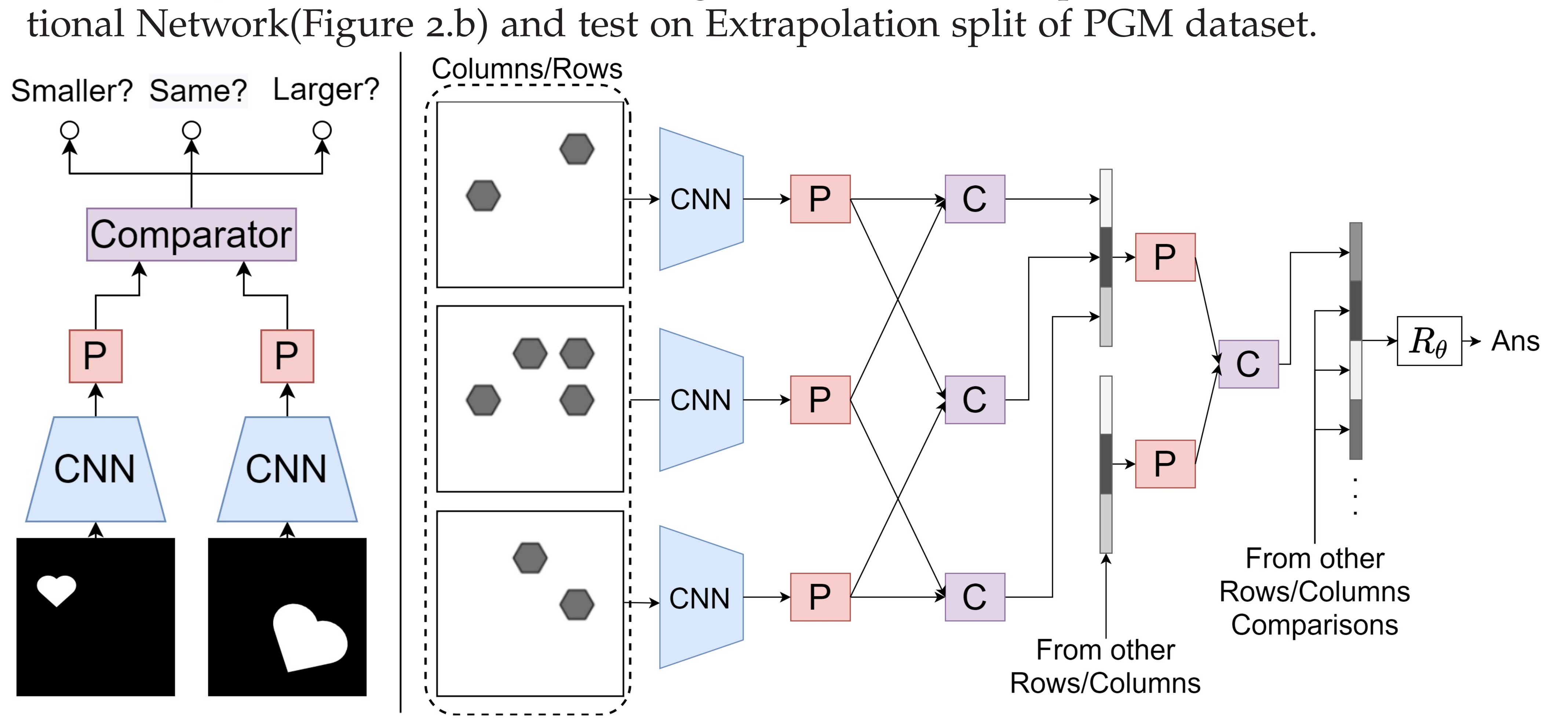
$$f(o_i, o_j) = g(\overset{K}{\underset{k=1}{||}} c_k(p_k(o_i), p_k(o_j))). \quad (1)$$

Here p_k is the k^{th} projection function that projects object representation o into the k^{th} low dimensional manifold, c_k is the k^{th} comparator function that compares the projected representations, $||$ is the concatenation symbol and g is a function that combines the K comparison results to make a prediction. Having K parallel projection functions p_k and comparators c_k enables a simultaneous comparison between objects with respect to their different attributes.

Relational Reasoning tasks and architectures

We amalgamate low-dim comparator into existing architectures, and achieved improved **o.o.d** generalization performance for three extrapolation relational reasoning tasks below. In order to test **o.o.d** generalisation in the extrapolation scenario, we create the training and test sets such that the ranges of compared attributes do not overlap.

- **Maximum of a set**: This goal of this task is to find the maximum of a set of real numbers. We amalgamate low-dim comparator into Set Transformer.
- **Visual Object Comparison**: the goal is to compare visual objects for different attributes such as size and spatial position. See Figure 2.(a) for details.
- **Raven Progressive Matrices**: We amalgamate low-dim comparators into a hierarchical Relational Network(Figure 2.b) and test on Extrapolation split of PGM dataset.



(a) (b)

2.(a) Visual Object Comparison (b) Raven Progressive Matrices.

Results

In this section we present extrapolation performance evaluation of low-dim comparators for the three relational reasoning tasks.

Maximum of a Set: For number sets for training, we uniformly sample numbers in the real value range $[0, 100)$. For testing we sample numbers in the range $[100, 200]$. We sampled 10000 sets for training and 2000 sets for testing. Table 1 shows the test error of our model compared to Deep Sets and Set Transformer, two previous state-of-the-art architectures for sets. Our model achieves a much lower extrapolation error than other methods, even lower than Deep Sets with a built-in Max-Pooling function.

Model	M.S.E
Deep Sets(Mean)	73.22 ± 17.11
Deep Sets(Max)	0.51 ± 0.29
Set Transformer	1.62 ± 0.76
OURS	$\mathbf{0.0015 \pm 0.0008}$

Table 1

Visual Object Comparison: we set three subtasks for comparing different attributes of the object, including size, horizontal position and colour intensity. Given the compared attribute range $[V_{low}, V_{high}]$, we sample training data from range $[V_{low}, \frac{2}{3}V_{high})$ and test data from range $[\frac{2}{3}V_{high}, V_{high}]$. For all experiments we sample 60000 training pairs and 20000 test pairs. We test our proposed model against an MLP baseline. Table 2 shows the extrapolation test accuracies of our model compared to the baseline. Our model with low-dimensional comparator significantly outperforms baselines for all three compared attributes.

Attributes	Baseline	OURS
Size	$79.52 \pm 6.71\%$	$\mathbf{94.05 \pm 3.03\%}$
X-Coord	$66.14 \pm 5.03\%$	$\mathbf{79.11 \pm 1.92\%}$
Colour	$78.45 \pm 3.34\%$	$\mathbf{97.71 \pm 2.81\%}$

Table 2

Raven Progressive Matrices: For RPM-style tasks we use the extrapolation split of the PGM dataset, which is already a well-defined extrapolation type of generalisation task. We compare our proposed model against all previous methods (to the best of our knowledge) that have reported results on the extrapolation data split. We additionally include a baseline model named "MLRN-P", which is a 2-layer MLRN with prior knowledge of only the relations present in rows/columns and with pre-training. Table 3 shows the test accuracy comparison. Our proposed model outperforms all other baselines. We note that we used a vanilla CNN as the perception module, same as most previous methods on RPM tasks.

Model	Accuracy
WReN	17.2%
MXGNet	18.9%
MLRN	14.9%
MLRN-P	18.1%
OURS	**25.9%**

Table 3

In Conclusion, our proposed low-dim comparator can be readily adapted into neural architectures to improve extrapolation performance for relational reasoning tasks.