TPR-Attention for Combinatorial Generalization

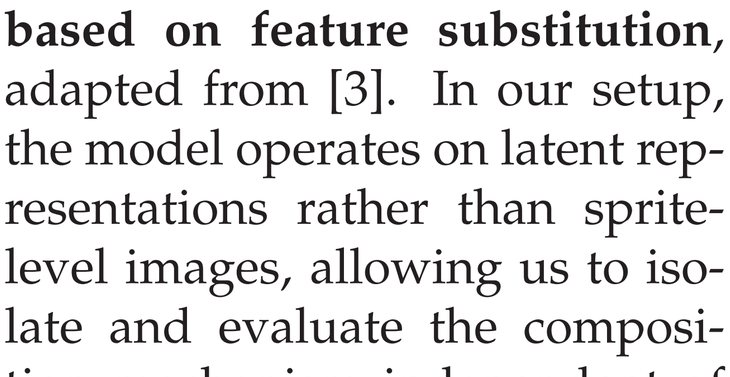
Melisa Civelekoğlu & Isabeau Prémont-Schwarz
Université Laval, Institut intelligence et données, Mila - Quebec AI Institute

Problem & Motivation

This research examines a **Tensor Product Representation (TPR)-based attention framework** to achieve **combinatorial generalization** – the ability to recombine known factors of variation. Achieving combinatorial generalization is crucial for many real-world decision-making problems, where it is implausible for data to cover all possible combinations.

Experimental Setup

We evaluate our mechanism on a **controlled composition task based on feature substitution**, adapted from [3]. In our setup, the model operates on latent representations rather than sprite-level images, allowing us to isolate and evaluate the composition mechanism independent of perceptual representation learning. We define **different out-of-distribution and interaction settings** based on **numerical and categorical factors of variation**.

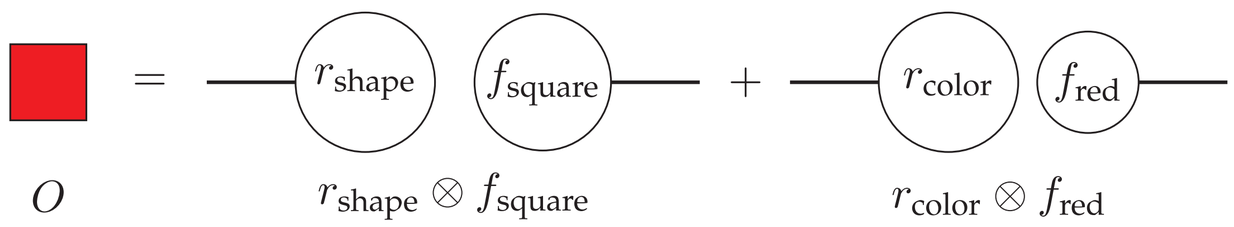


References

[1] Emile Mathieu, Tom Rainforth, N. Siddharth, and Yee Whye Teh. Disentangling disentanglement in variational autoencoders, 2019.
[2] Xin Wang, Hong Chen, Si'ao Tang, Zihao Wu, and Wenwu Zhu. Disentangled representation learning, 2024.
[3] Milton L. Montero, Jeffrey S. Bowers, Rui Ponte Costa, Casimir J. H. Ludwig, and Gaurav Malhotra. Lost in latent space: Disentangled models and the challenge of combinatorial generalisation, 2024.
[4] Paul Smolensky. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, 46(1):159–216, 1990.

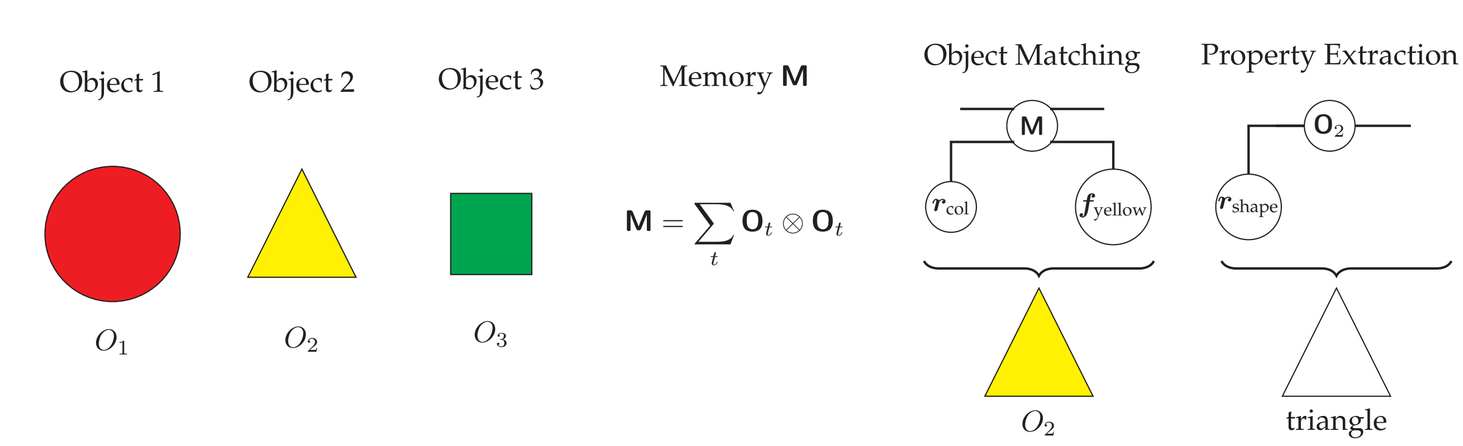
Background

Previous works focus on learning disentangled representations [1, 2]. However, evidence suggests that disentanglement alone may not guarantee compositional generalization [3]. Montero et al. (2024) show that models achieving high disentanglement can still fail to generalize compositionally when factors interact [3]. **Our work focuses on this hard case of interacting factors of variation**.
A TPR encodes structured objects using role-filler bindings [4]. **Roles** specify an attribute slot (e.g., *shape*, *colour*), and **fillers** are the values occupying each role (e.g., *square*, *red*).



Methodology

There are three stages in TPR-Attention: (i) generate a structured role-filler query and match relevant objects, (ii) extract a target property from each matching object and (iii) transform each extracted property and re-bind it to a new role per attention head.



To match and extract at the same index i, we define a TPR-based structured associative memory mechanism (TPR-SAM):

$$\mathbf{M}_t \leftarrow \mathbf{M}_{t-1} + \mathbf{O}_t \otimes \mathbf{O}_t = \sum_{s=1}^{t} \mathbf{O}_s \otimes \mathbf{O}_s$$

The extracted property from the matched object is then transformed by a learned matrix $\boldsymbol{H}$ and re-bound to a new role $\boldsymbol{r}_n$ per attention head.

Results - Loss on Combinatorial Generalization