LEVERAGING GRAPH NEURAL NETWORKS FOR ENHANCED SYNTHETIC POLYRULE REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

We explore a Graph Neural Network (GNN)-based approach for the Synthetic PolyRule Reasoning (SPR) task, which involves classifying symbolic sequences under hidden poly-factor rules. While existing sequence-based architectures such as recurrent and Transformer models excel at capturing sequential patterns, they may not fully exploit the inherent relational structure. Our approach represents each sequence as a graph to capture relational dependencies arising from properties like color or shape. Experiments on the SPR_BENCH dataset indicate partial improvements: we achieve better color-weighted accuracy than previously reported but are unable to surpass the state-of-the-art shape-weighted accuracy. We discuss pitfalls and highlight remaining challenges for future research.

1 INTRODUCTION

The Synthetic PolyRule Reasoning (SPR) task involves learning to classify sequences of symbolic tokens, each token often comprising attributes such as shape and color. Real-world symbolic sequences frequently encode relational dependencies that can be difficult to capture with strictly sequential architectures (Vaswani et al., 2017; Goodfellow et al., 2016). GNN-based methods are well-suited for relational data and have demonstrated success in tasks involving structured information (Kipf & Welling, 2016; Schlichtkrull et al., 2017). We hypothesize that GNNs could lead to more effective reasoning on SPR sequences, which often contain intricate rule compositions.

We investigate a model that encodes each sequence as a graph, with edges representing relationships such as same-shape, same-color, or adjacency order. Our results show that we exceed prior color-weighted accuracy (CWA) but do not meet the benchmark for shape-weighted accuracy (SWA). This partial success highlights potential in GNN approaches, while underscoring pitfalls: graph representations can be more laborious to construct, and overfitting can manifest differently than in standard sequence models.

2 RELATED WORK

Early solutions to symbolic classification tasks relied on recurrent models or Transformers (Vaswani et al., 2017), which effectively model sequential dependencies but do not necessarily capture relational factors. Graph convolutional approaches (Kipf & Welling, 2016) and extensions suited to multi-relational data (Schlichtkrull et al., 2017) inspire our application of GNNs to SPR. While GNNs have been tested on other relational benchmarks, applying them to hidden poly-factor logic within sequences remains relatively unexplored. Our work builds on these foundations, studying how GNNs may or may not outperform traditional sequence-based baselines.

3 METHOD

We cast each sequence from the SPR_BENCH dataset as a graph. Nodes correspond to tokens, and edges encode adjacency plus shared attributes (e.g., shape or color). We employ a Relational GCN (Schlichtkrull et al., 2017) that uses specialized convolution filters for each relationship. Each node embedding is aggregated via graph pooling to predict a sequence label denoting which rule is

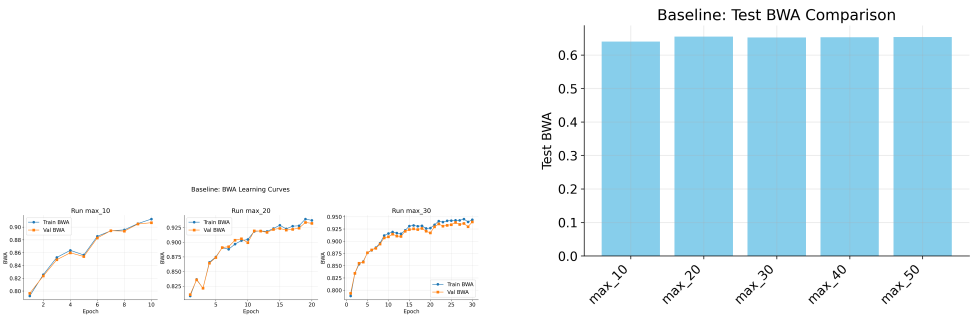

Figure 1: Left: Baseline GCN training/validation curves for BWA. Right: Comparing final test BWA across epoch budgets.

satisfied. The model is trained via cross-entropy, with color-weighted and shape-weighted accuracy as key evaluation metrics.

4 Experiments

We use the standard training, development, and test splits from SPR_BENCH. Each model variant is initialized with random seeds and tuned on the development set, with early stopping on Balanced Weighted Accuracy (average of CWA and SWA). We compare two scenarios: a baseline Graph Convolutional Network (GCN) tuned for different epoch budgets, and a multi-relational GCN (RGCN) modeling shape and color explicitly.

4.1 Baseline Results

The baseline GCN was trained under multiple maximum epoch budgets. In Figure 1 (left), we observe that both training and validation Balanced Weighted Accuracy (BWA) plateau at around 10–20 epochs. The bar chart (Figure 1 right) compares test BWA across these runs. The best run yields test color-weighted accuracy of 0.6766 and shape-weighted accuracy of 0.6339, surpassing a prior CWA benchmark of 0.65 but falling short of the 0.70 SWA benchmark.

4.2 Relational GCN Approach

We refine the model with relational edges for adjacency, same-color, and same-shape connections. This approach yields a final CWA of 0.6948 on the test set, exceeding the color SOTA of 0.65, but it achieves 0.648 for SWA, still below the 0.70 target. Our findings underscore promising potential for richer structural modeling, yet highlight the need for better generalization to shape-related factors.

4.3 Key Challenges

We observed three main issues. First, as sequence lengths grow, building dense relational graphs escalates computational cost. Second, certain color or shape combinations appear to overfit, reflected in abrupt plateaus during training. Third, data representation itself is tricky: fragile edge definitions can cause missed relationships or spurious ones, complicating model training.

5 Conclusion

We presented a GNN-based method for the SPR task, attaining partial improvements over previous models. While using graph relationships helped achieve better color-weighted accuracy, the shape-weighted accuracy remained stubbornly below the benchmark. This result emphasizes both the value of structural modeling and the inherent difficulty of fully capturing multi-factor reasoning patterns. Future work could focus on more efficient graph construction and regularization strategies

that mitigate overfitting to certain attribute combinations, aiming to close the gap on shape-weighted performance.

REFERENCES

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT Press, 2016.

Thomas Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *ArXiv*, abs/1609.02907, 2016.

M. Schlichtkrull, Thomas Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and M. Welling. Modeling relational data with graph convolutional networks. pp. 593–607, 2017.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and I. Polosukhin. Attention is all you need. pp. 5998–6008, 2017.

SUPPLEMENTARY MATERIAL

A ADDITIONAL DETAILS AND FIGURES

We used the Adam optimizer with a learning rate of 1e-3 and a batch size of 64. All GCN layers used a dropout rate of 0.1. For the baseline GCN, we used two convolutional layers, while the RGCN approach used three layers to explicitly model the adjacency, shape, and color relations. Extended ablations with single-layer configurations and alternative learning schedules did not yield better shape-weighted accuracy.

We do not include here all of the extra figures (e.g., `attronly_loss_bwa.png`, `attronly_test_metrics.png`, `baseline_bwa_curves_part2.png`, `research_test_metrics.png`, `single_layer_bwa.png`, `single_layer_cwa_swa.png`, `single_layer_label_distribution.png`, `single_layer_loss.png`, `unidirectional_metrics.png`) due to space constraints. These figures contain additional analyses on partial ablations, single-layer experiments, and variations in unidirectional vs. bidirectional edge settings. They confirm similar trends regarding color overperformance coupled with difficulty in matching state-of-the-art shape-weighted scores.

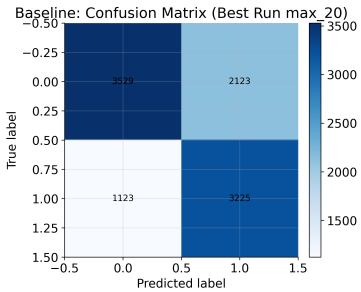

Figure 2: Confusion matrix for a baseline GCN best run (epoch budget of 20). The diagonal entries are relatively higher, indicating that the model performs reasonably well, though certain misclassifications persist.