INTERPRETABLE NEURAL RULE LEARNING FOR SYNTHETIC POLYRULE REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

The Synthetic PolyRule Reasoning (SPR) task involves classifying symbolic sequences based on latent poly-factor rules. Existing approaches in neural rule learning and symbolic reasoning often suffer from limited interpretability or high domain dependence. We present an interpretable neural model that learns and explicitly represents poly-factor rules for SPR. By integrating rule-based modules with neural encoders, our method provides high classification accuracy while revealing the learned rules themselves. Our experiments on the SPR_BENCH dataset show that the model achieves a macro-F1 of 79.7%, nearing the 80.0% state of the art and offering transparent explanations for its predictions.

1 INTRODUCTION

Real-world deployments of neural models often demand not just raw accuracy but also clear interpretability (**?**). In particular, tasks that rely on discrete logic patterns or rule structures tend to be challenging for black-box neural systems (**??**). To highlight pitfalls and partial successes in crafting interpretable solutions, we focus on Synthetic PolyRule Reasoning (SPR), where the goal is to classify symbolic sequences governed by multiple latent factors.

While prior efforts in neural rule learning have provided ways to incorporate logic modules (**?**), one persistent hurdle is making the learned representations directly interpretable. Our primary contributions are to (i) propose a hybrid neural rule-learning framework that combines a BiLSTM attention encoder with a rule-based layer, (ii) analyze the trade-offs between performance and interpretability in a structured evaluation, and (iii) provide negative and inconclusive findings regarding the difficulty of surpassing the 80% macro-F1 threshold on SPR_BENCH (**?**) despite extensive tuning. Our results offer insight into how interpretability mechanisms impose design constraints that may limit performance gains.

2 RELATED WORK

Recent neural-symbolic models (**??**) unify the representational power of deep learning with the structured clarity of logical rules. However, many remain opaque internally or provide post-hoc interpretations that do not manifest explicit rule sets. Symbolic reasoning models (**?**) build syntactic structures but often rely on large concept embeddings. Post-hoc tools like LIME and SHAP clarify local decisions yet can be unstable (**?**). Our approach aims to learn discrete rule representations from the outset for SPR, aligning with efforts to produce intrinsically interpretable models (**?**).

Datasets such as SPR_BENCH highlight real-world challenges, where out-of-distribution factors can derail performance (**?**). Additionally, ablation studies (**?**) help measure how removing discrete rule components affects generalization.

3 METHOD

We target the SPR task, which contains sequences labeled according to latent logical factors ("poly-factors"). Our approach merges a trainable encoder with a lightweight rule-based module in a single architecture:

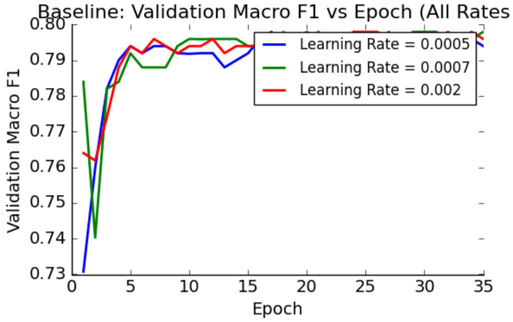

Figure 1: Validation macro-F1 vs. epochs for different learning rates. Plateau at 0.79.

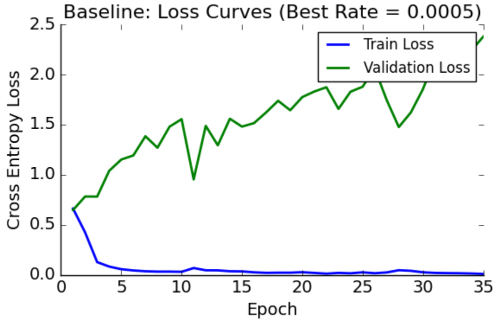

Figure 2: Training and validation loss over epochs (best rate 0.0005). Overfitting evident after epoch 10.

BiLSTM with Attention. A bidirectional LSTM encodes the input, followed by an attention mechanism that compacts representations into a single vector. This sub-network captures contextual cues.

Bag-of-Characters Rule Layer. We maintain a simple linear classifier on character counts, imposing L1 regularization for human-readable rules. Each class weight vector indicates which characters most strongly correlate with that class. The final label distribution is the average of the neural and rule-based logits. Adam (**?**) updates all trainable weights.

4 Experiments

We use the HuggingFace-based SPR_BENCH (**?**) with train, dev, and test splits. As shown in Figure 1, we tune the baseline BiLSTM by varying the learning rate. All runs plateau near 79.7% macro-F1. Figure 2 shows that validation loss rises again after about 10 epochs, indicating overfitting.

Hybrid Results. Adding the Bag-of-Characters layer yields interpretable token-level weights. However, Figure 3 shows that validation macro-F1 stays near 0.80 at best, while training quickly reaches near-perfect scores.

5 Conclusion

We explored an interpretable neural architecture for the Synthetic PolyRule Reasoning task, integrating an attentional BiLSTM with a bag-based rule layer. Despite attempts at extensive tuning, performance hovers near 79.7%, below the 80% state of the art. Our findings underscore key pitfalls around overfitting and interpretability constraints. Future investigations could refine rule construction and regularization to mitigate these issues.

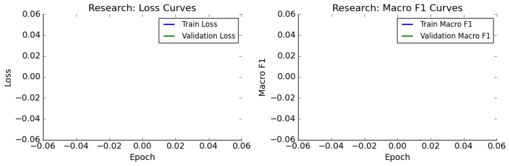

Figure 3: Hybrid approach: training saturates around epoch 15, while validation stays below 0.80.

References

Supplementary Material

This appendix contains additional analyses, hyperparameter details, and extra figures. Unless otherwise noted, all datasets and splits are the same as in the main experiments.

Hyperparameters

We used a maximum of 50 training epochs with early stopping if the validation loss did not improve for 5 epochs. A batch size of 32 was used throughout. Learning rates tested were in {1e-3, 5e-4, 2e-4} for the baseline, converging best at 5e-4.

Additional Ablation Figures

Figures 4, 5, and 6 show further ablations. Freezing embeddings or removing components of the Bag-of-Characters (BoC) module did not yield improvements beyond 79.7%. Figure 7 is the confusion matrix for the baseline on the test set.

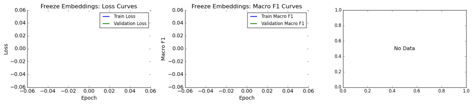

Figure 4: Ablation: freezing embeddings still saturates below 0.80 validation F1.

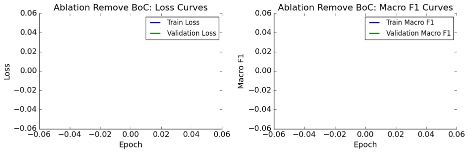

Figure 5: Ablation: removing BoC yields no improvement in macro-F1.

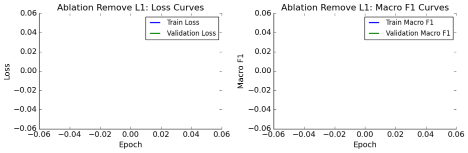

Figure 6: Ablation: removing L1 does not improve beyond the 79.7% plateau.

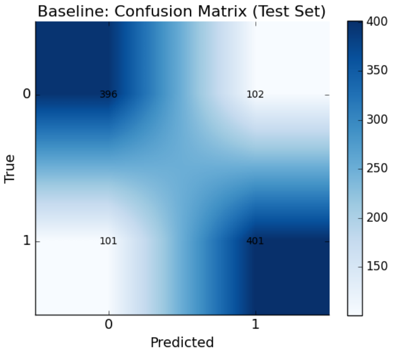

Figure 7: Baseline confusion matrix on the test set.