ENHANCING TRANSFORMER MODELS WITH SYMBOLIC REASONING CAPABILITIES FOR SYMBOLIC POLYRULE REASONING

Anonymous authors
Paper under double-blind review

ABSTRACT

We investigate the conceptual generalization capabilities of transformer models on a symbolic classification task we call Symbolic PolyRule Reasoning (SPR). SPR involves sequences of abstract symbols whose labels depend on hidden poly-factor rules. We hypothesize that augmenting transformers with explicit symbolic modules can preserve overall accuracy close to state-of-the-art levels while improving interpretability and rule-based reasoning. We train baseline transformers of varying depth and compare them with a hybrid neural-symbolic model that integrates a symbolic head. We observe that all models reach around 70% test macro-F1, but exhibit overfitting and limited systematic generalization. Our results highlight the real-world pitfalls of relying on sub-symbolic pattern matching when explicit rule-based inference is needed.

1 INTRODUCTION

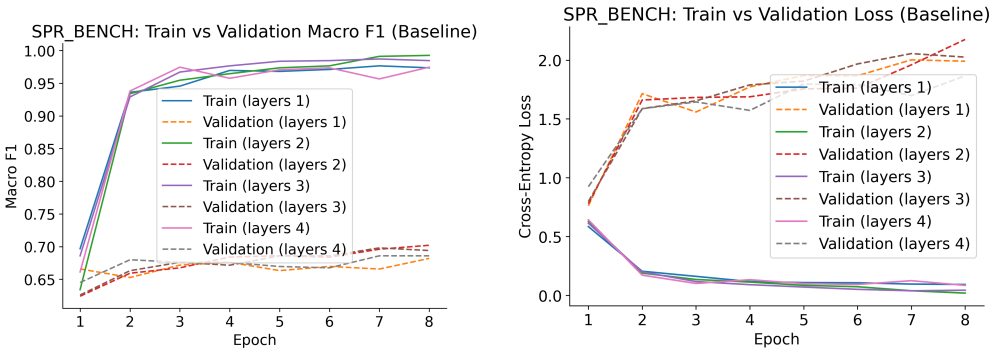
Symbolic reasoning tasks challenge neural networks to extrapolate beyond familiar patterns. Yet, in real-world scenarios, lacking robust rule-grounded generalization can cause unreliability when new patterns arise. By integrating explicit symbolic modules, researchers hope to endow models with deeper logical inference and interpretable decision-making. Still, important questions remain as to whether these additions practically mitigate errors under distribution shifts.

Our work centers on a new task, Symbolic PolyRule Reasoning (SPR), in which sequences of abstract tokens are classified according to hidden poly-factor generation rules. This is reminiscent of multi-step reasoning challenges (Patel et al., 2024; Xu et al., 2024; Pung & Chan, 2021), where unseen factor combinations demand structured generalization. Transformer-based architectures (Vaswani et al., 2017) frequently excel on pattern recognition but often fail to extrapolate systematically (Bergen et al., 2021). Neural-symbolic frameworks (Garcez et al., 2015) suggest unifying sub-symbolic and symbolic logic, though empirical demonstrations of expanded real-world reliability remain limited.

In this paper, we introduce SPR and highlight three main contributions: (1) A dataset to test rule-based classification under unseen factor combinations, (2) Comparisons between standard transformers of varying depth and a hybrid model merging learned embeddings with a symbolic head, (3) Empirical evidence of systematic overfitting, with all models saturating near 70% macro-F1 despite perfect training accuracy. Our findings underscore the practical importance of building architectures that truly capture underlying rules rather than memorizing surface patterns.

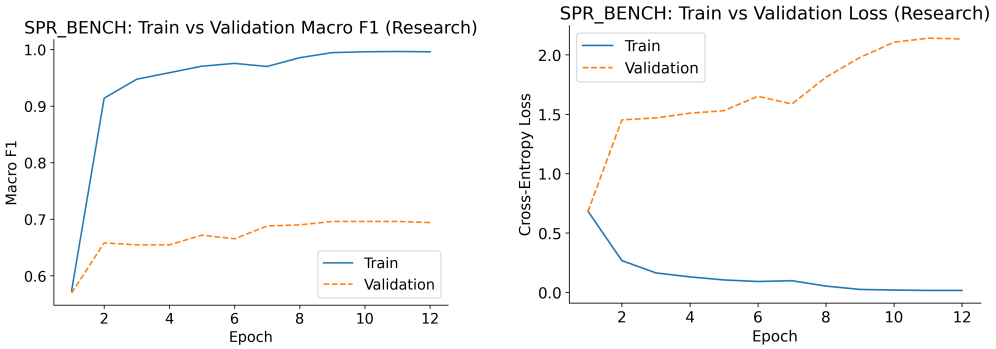
2 RELATED WORK

Many studies have shown that neural models often memorize training artifacts rather than learn robust rules (Bergen et al., 2021). Extrapolation failures appear in specialized benchmarks like ORCHARD (Pung & Chan, 2021) and Multi-LogiEval (Patel et al., 2024), revealing how distribution shifts break naive models. Despite the promise of neural-symbolic methods (Garcez et al., 2015), examples of significant gains in real scenarios are still scarce. Our SPR dataset builds on these lines by requiring multi-factor logic under controlled but challenging conditions.



(a) Training vs. Validation Macro-F1

Figure 1: **Baseline transformer performance by depth.** (Left) Macro-F1 for training (solid) and validation (dashed) regimes, showing rapid overfitting. (Right) Cross-entropy loss curves mirror the F1 patterns. Despite near-perfect train metrics, validation saturates near 0.70.



(a) Training vs. Validation Macro-F1

Figure 2: **Neural-symbolic hybrid results.** Adding a symbolic head does not substantially improve validation F1 (dashed), which remains capped at about 0.70. Training performance saturates near 0.99, indicating a similar overfitting pattern.

3 Method and Experiments

SPR Task. We construct 20k/5k/10k splits of sequences up to length 64, labeled by hidden, poly-factor classification rules (*class0* or *class1*). Besides standard train/test evaluations, we measure systematic generalization accuracy (SGA) on bigram patterns withheld from training.

Models. We train transformer encoders with 1–4 layers (Vaswani et al., 2017), applying self-attention over token embeddings. Our hybrid model appends a bag-of-symbols feature vector to the transformer's final hidden state. Both approaches use the Adam optimizer (learning rate 1×10^{-4}, batch size 128) with 2k-step warmup, standard dropout (0.1), and 20 training epochs.

Results and Analysis. Figure 1 (left) demonstrates how training macro-F1 quickly rises to above 0.99 for baseline transformers, while validation F1 levels around 0.70, indicative of overfitting. The loss curves (right) confirm the same pattern: deep layers can fit training data well but fail to generalize. Figure 2 similarly shows the neural-symbolic hybrid saturating near 0.70 on validation. These plots illustrate how simply adding a symbolic feature does not substantially mitigate the distribution shift challenge posed by unseen factor combinations.

Overall, the consistent gap between near-perfect training performance and suboptimal test or withheld bigram accuracy highlights the fragility of purely sub-symbolic approaches. Even explicit symbolic features have limited impact unless the system properly encodes and applies the underlying logic.

4 CONCLUSION

We introduced the Symbolic PolyRule Reasoning challenge to test rule-based classification under unseen factor combinations. Baseline transformers and a neural-symbolic hybrid all converge to roughly 70% macro-F1, exposing critical weaknesses in systematic generalization. These observations reinforce the urgency of designing architectures or training protocols that genuinely capture underlying logical structure. We hope our results spur further exploration into robust, rule-aware solutions that perform reliably beyond the training distribution.

REFERENCES

Leon Bergen, T. O'Donnell, and Dzmitry Bahdanau. Systematic generalization with edge transformers. pp. 1390–1402, 2021.

A. Garcez, Tarek R. Besold, L. D. Raedt, Peter Földiák, P. Hitzler, Thomas F. Icard, Kai-Uwe Kühnberger, L. Lamb, R. Miikkulainen, and Daniel L. Silver. Neural-symbolic learning and reasoning: Contributions and challenges. 2015.

Nisarg Patel, Mohith Kulkarni, Mihir Parmar, Aashna Budhiraja, Mutsumi Nakamura, Neeraj Varshney, and Chitta Baral. Multi-logieval: Towards evaluating multi-step logical reasoning ability of large language models. *ArXiv*, abs/2406.17169, 2024.

B. Pung and Alvin Chan. Orchard: A benchmark for measuring systematic generalization of multi-hierarchical reasoning. *ArXiv*, abs/2111.14034, 2021.

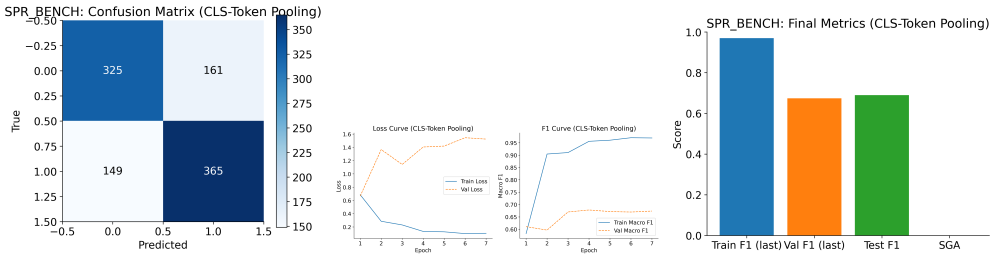
Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and I. Polosukhin. Attention is all you need. pp. 5998–6008, 2017.

Jundong Xu, Hao Fei, Liangming Pan, Qian Liu, M. Lee, and W. Hsu. Faithful logical reasoning via symbolic chain-of-thought. pp. 13326–13365, 2024.

SUPPLEMENTARY MATERIAL

A ABLATION AND ADDITIONAL RESULTS

We conducted ablation studies on specific architectural choices (e.g., removing the `[CLS]` token, positional encodings) to investigate whether slight design changes could improve generalization. Our findings remain consistent: all models still overfit training data and fail to systematically extrapolate. Figure 3 shows a sample of these results, combining confusion matrices, learning curves, and final metric summaries for one ablation variant. Despite minor differences, neither removing `[CLS]` tokens nor changing positional encodings yielded substantial improvements.

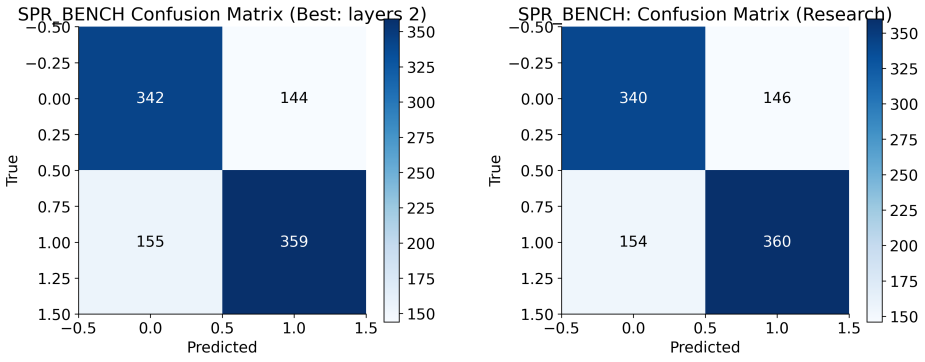


(a) Confusion Matrix (b) Training & Validation Curves (c) Final Metrics

Figure 3: **Ablation without `[CLS]` token.** (a) Confusion matrix shows a mix of correct and incorrect predictions, consistent with ∼70% F1. (b) Clear overfitting persists: training saturates, validation stalls. (c) Summary metrics confirm that removing `[CLS]` tokens does not alleviate the shortfall.

B BASELINE VS. HYBRID CONFUSION MATRICES

Outside of single-number metrics, confusion matrices can highlight the nature of misclassification. Figure 4 compares baseline and hybrid predictions on the same test set. While the distributions of true positives/negatives and errors differ slightly, both approaches struggle with sequences containing novel symbol pairs that violate training distributions.



(a) Baseline (b) Hybrid

Figure 4: **Comparison of confusion matrices.** Neither the baseline nor the hybrid approach achieves robust classification on withheld combinations, as reflected in frequent off-diagonal predictions.

C REMOVED OR UNUSED FIGURES

We omit certain unused ablation and test F1 bar charts that added little additional insight. For completeness, we note that removing positional encodings or using symbols-only modules yielded similar training overfits and validation plateaus, thus we do not replicate them here. Interested readers may contact the authors for raw logs and additional plots.

D IMPLEMENTATION DETAILS

All experiments are implemented using PyTorch with standard transformer modules. We use Adam with a learning rate 1×10^{-4}, batch size 128, dropout 0.1, and 20 total epochs. Hyperparameters were tuned lightly on the dev set. Additional training details or parameters (e.g., random seeds, hardware configuration) are identical across baselines and ablation variants.