A Microscopic Study of Decimal Parity in a Tiny Causal Transformer

SnurfyAI Research Team

May 2026

Abstract

We study a 217,664-parameter GPT-style sequence classifier trained to decide whether a decimal numeral is even or odd from a ChatML-formatted prompt. Although the task is locally solvable from the final digit alone, the model offers a compact setting for studying how tiny transformers acquire simple algorithmic rules, how robustly they generalize across length, and how attention summaries relate to causal feature importance. On a fresh balanced benchmark of 100,000 random numbers of length 1–249, the evaluated checkpoint reaches 99.97% accuracy. A length-stratified sweep shows no systematic degradation near the context limit. Fixed-prefix digit sweeps reveal perfect rule-consistent behavior when only the last digit is changed, while counterfactual interventions show that flipping the parity of the last digit changes the prediction essentially every time, whereas randomizing the prefix almost never does. Head-level attention analysis shows a mixed picture: one intermediate head often points to the last digit, but later layers distribute much more attention over the final three-digit suffix region than over the final digit alone. This does not contradict the intervention results, since attention weights are not direct measures of feature importance. Training dynamics further show a delayed plateau-then-jump transition, but not a clean demonstration of canonical grokking. Overall, the results show that a tiny transformer can learn the decimal parity rule very robustly, even though the underlying computations are more distributed than a simple final-digit lookup would suggest.

1 Introduction

Small algorithmic tasks are useful microscopes for studying generalization and internal computation in neural networks. They are simple enough to evaluate exhaustively or nearly exhaustively, yet rich enough to expose nontrivial optimization dynamics and representational structure. Prior work on grokking and modular arithmetic has shown that neural networks can move from memorization to crisp generalization late in training, and that compact transformers can implement interpretable arithmetic circuits under the right conditions [1, 2, 3, 5].

This paper studies an even smaller setting: decimal parity classification. Given a decimal numeral $x = d_1 d_2 \dots d_n$, the correct label is determined by the final digit alone:

$$\mathrm{even}(x) = \mathbf{1}\{d_n \in \{0, 2, 4, 6, 8\}\}.$$

Unlike the canonical PARITY problem on binary strings, which depends on all input bits and is a central object in transformer expressivity theory [4], decimal parity is a local rule. That makes the task easier, but it also makes the setting useful for a different question: when a transformer should be able to solve a task by attending to one decisive symbol, does it actually do so?

The model generalizes extremely well, though not perfectly, to fresh random numbers. Its behavior is causally dominated by the last digit: changing only the last digit flips the decision,

whereas changing the prefix rarely affects the outcome. The attention maps suggest a more nuanced pattern. Averaging across heads obscures a head that frequently attends to the last digit, while later layers focus heavily on the final few digits as a suffix neighborhood. In addition, the training history reveals a delayed transition from near-chance to near-perfect performance despite the simplicity of the target rule.

The result is a compact case study in which behavioral correctness, causal feature importance, and raw attention are related but not identical. Model weights and further release information are available at `https://hf.co/SnurfyAI/is-even`; future training-code release details will be maintained on the model card.

2 Task and Experimental Setting

2.1 Model and data

The model is a decoder-only transformer sequence classifier with 4 transformer layers, 4 attention heads per layer, embedding size 64, and 217,664 trainable parameters. Tokenization is strictly digit-level. The vocabulary contains 17 tokens: the ten decimal digits, the labels `true` and `false`, four ChatML control tokens, and a padding token. With the prompt template included, the 256-token context window supports inputs up to 249 digits.

Training data are generated syntheticallyFor each digit length from 1 to 249 and each final digit from 0 to 9, we generate 3 training examples and 1 validation example, for 7,470 training examples and 2,490 validation examples total. All other digits are sampled uniformly at random subject to the no-leading-zero constraint. The Model was trained for 60 epochs using AdamW, batch size 64, learning rate 10^{-3}, and linear warmup-decay.

2.2 Evaluation protocol

We report five analyses.

1. A fresh large-sample benchmark on 100,000 balanced random examples with lengths drawn uniformly from 1 to 249.
2. A length-stratified sweep with 40 examples at each supported digit length.
3. Fixed-prefix digit sweeps in which only the last digit changes.
4. Counterfactual interventions that separately perturb the prefix and the final digit.
5. Attention analysis using the final `assistant` token as the decision position, with both layer-level averages and head-level breakdowns, and with normalization both over digit positions and over all tokens.

3 Related Work

Our study sits between four nearby literatures.

First, *grokking* work studies delayed generalization on small algorithmic datasets. Power et al. [1] introduced the phenomenon on modular arithmetic and showed that perfect generalization can emerge long after overfitting. Gomezjurado Gonzalez [5] recently argued that, in arithmetic tasks, the delay can come from limited access to already learned structure rather than from the absence of structure itself.

Table 1: Core behavioral results for the trained model.

Experiment	Examples	Accuracy	Main observation
Fresh random benchmark	100,000	0.9997	30 errors; 29 are odd → even
Length-stratified sweep	9,960	0.9999	One failure; no systematic length trend
Fixed-prefix digit sweeps	10,000	1.0000	Changing only the last digit follows parity exactly

Second, *mechanistic studies of arithmetic transformers* analyze how compact models implement algorithmic tasks. Furuta et al. [2] studied grokked transformers on modular polynomials using Fourier analysis and ablation. Africa et al. [3] examined modular exponentiation and reported specialized attention-head subgraphs sufficient for full task performance. These studies motivate our head-level analysis: coarse layer averages can hide sparse, task-specific heads.

Third, a separate *attention interpretability* literature argues that attention weights should not be read as direct explanations or causal feature-importance scores [6]. Our paper does not try to overturn that point; instead, it uses decimal parity as a compact example in which attention summaries and causal interventions answer different questions.

Fourth, *theoretical work on parity and transformer expressivity* addresses a much harder parity setting than ours. Kozachinskiy et al. [4] study binary PARITY, prove lower bounds for one-layer transformers, and construct parity-computing transformers under realistic softmax and masking assumptions. Our decimal even/odd task is simpler because the last digit is sufficient. This difference is important: our contribution is not to solve the expressivity question for general parity, but to characterize how a tiny transformer behaves when the correct rule is local and symbol-based.

To our knowledge, we did not identify prior work focused specifically on decimal parity classification in a tiny decoder-only transformer trained on chat-formatted synthetic data. The closest prior studies instead address binary parity, modular arithmetic grokking, or more complex arithmetic tasks.

4 Results

4.1 Generalization is excellent but not literally perfect

Table 1 summarizes the core behavioral results. On 100,000 fresh balanced random examples, the model achieves 99.97% accuracy. Errors are rare but real: 30 total mistakes, consisting of 29 false positives on odd numbers and 1 false negative on an even number. The error distribution is asymmetric and concentrated on odd endings, especially 7, 5, and 1.

The length-stratified sweep yields 99.99% accuracy over 9,960 examples, with only one observed failure and no meaningful trend toward degradation at long input lengths. In particular, 249-digit examples remain perfectly classified in the length sweep.

These results are strong enough for a paper-scale case study. They are also useful precisely because they are not trivially perfect: the residual failures let us study both robustness and the gap between rule-like behavior and flawless symbolic execution.

The residual errors are not uniformly distributed. Of the 30 mistakes in the 100,000-example benchmark, 23 fall in the 10–99 digit range, 3 in the 100–199 range, and 4 in the 200–249 range; none occur on 1–9 digit inputs. By final digit, 7 accounts for 10 errors, 5 for 7, and 1 for 6, while only a single even-ending input fails at all. This strengthens the impression that the model has learned an almost rule-based solution with a small remaining bias toward predicting the positive class.

Table 2: Counterfactual sensitivity of the model. Prediction-change rates are measured relative to the original example. The conditional column restricts to examples that were correctly classified before intervention.

Intervention	Accuracy	Change rate	Change rate given base correct
Randomize prefix, keep last digit	0.9998	0.0006	0.0002
Replace last digit, same parity	0.9998	0.0002	0.0000
Replace last digit, opposite parity	1.0000	0.9996	1.0000
Replace one random prefix digit	0.9996	0.0000	0.0000

Table 3: Digit-only attention from the decision token to the suffix.

Layer	Last-digit share	Last-three-digit share
0	0.0119	0.0583
1	0.0199	0.0590
2	0.0247	0.2319
3	0.0188	0.2551
Uniform baseline	0.0242	0.0622

4.2 The model has learned the correct causal feature

The clearest mechanism result comes from interventions. For 1,000 random prefixes, we swept the final digit through 0–9 while holding the prefix fixed. The model followed the parity rule perfectly in all 10,000 such examples. This establishes that the learned computation is strongly controlled by the final digit.

We then compared four larger counterfactual families over 5,000 balanced random base examples. The results appear in Table 2.

The qualitative pattern is decisive. Randomizing the entire prefix while preserving the last digit almost never changes the prediction. Replacing one random prefix digit never changes it in our 5,000-example study. Replacing the last digit with another digit of the *same* parity also leaves the decision unchanged. But replacing the last digit with one of *opposite* parity flips the prediction for every base-correct example and for 99.96% of all examples overall.

This is strong evidence that the model has learned the right causal rule: parity is encoded primarily through the suffix, and especially through the parity class of the final digit.

4.3 Attention summaries and causal importance diverge

Behaviorally, the last digit is the decisive feature. Internally, the model is more distributed than that description suggests. This is not paradoxical. Prior interpretability work has emphasized that attention weights are not direct explanations or causal importance maps [6]. Our goal here is therefore narrower: to characterize what attention pattern accompanies a classifier whose causal behavior is overwhelmingly determined by the final digit.

We measured attention from the final `assistant` token—the position whose hidden state feeds the classification head—to preceding digit positions on 1,000 balanced random examples.

Table 3 reports the mean share of digit-only attention assigned to the last digit and to the last three digits. For reference, uniform attention over all digits would allocate 2.42% to the final digit and 6.22% to the final three digits on this sample.

Two points matter.

First, the last digit is *not* the dominant attended digit on average. Its mean share is at or below the uniform baseline in most layers, and in the later layers the most attended suffix positions are usually the penultimate and antepenultimate digits. Second, the suffix as a *region* is highly salient in late layers: the last three digits receive about 23–26% of digit-only attention in layers 2 and 3, roughly four times the uniform baseline.

Head-level analysis adds an important refinement. Averaging across heads does hide one sparse pointer-like component: in layer 1, head 2 takes the last digit as its top-attended digit in 73.8% of examples. However, no later-layer head shows comparable last-digit top-1 behavior: the best rates are 22.3% in layer 2 and 5.7% in layer 3. The qualitative picture also survives a less favorable normalization. When control tokens are included rather than excluded, the mean last-digit share remains small across layers (0.61%, 1.50%, 1.93%, and 1.48% from layers 0 to 3).

We therefore avoid the stronger claim that the model lacks any pointer-like computation. The evidence is instead consistent with a mixed mechanism: at least one intermediate head frequently points to the last digit, while later layers pool information over the suffix neighborhood. Distinguishing this from a two-step operation based on positional cues and value-vector routing would require stronger causal tracing than we provide here.

4.4 Training shows delayed rule acquisition, but not canonical grokking

The saved 60-epoch training history shows a conspicuous phase transition. Validation accuracy remains between roughly 0.50 and 0.61 for the first 12 epochs, while training loss is still relatively high throughout that period (minimum 0.616). Validation then rises rapidly from epoch 13 onward, reaching 0.90 and 0.95 at epoch 17, 0.99 at epoch 25, and 1.00 at epoch 30. Stable saturation above 0.999 begins at epoch 29, leaving 31 later epochs in an already solved regime.

This is reminiscent of grokking, but it is not a clean demonstration of canonical memorization-then-generalization. We do not observe evidence that the model first reaches near-perfect training fit while validation remains near chance. Instead, the available evidence is more conservative: the model experiences a long plateau and then acquires a strongly generalizing rule rather abruptly. A fresh confirmatory 60-epoch retraining run reproduces the original near-perfect trajectory, while shorter alternative schedules underperform substantially. The important caveat is that changing the epoch count changes the linear learning-rate schedule itself, so shorter runs are not prefixes of the successful 60-epoch trajectory. We therefore describe this behavior as delayed rule acquisition or a plateau-then-jump regime with only a qualitative resemblance to grokking.

5 Discussion

The paper's main claim is narrow. We do *not* study the full binary PARITY problem, and we do not claim a new expressivity result. Decimal parity is much easier because the correct label depends only on the last digit. That simplicity is exactly why the case study is useful.

First, it provides a concrete example of a known interpretability caution: strong causal dependence need not appear as dominant raw attention to a single token. The last digit controls the classifier under counterfactual perturbations, yet later layers focus even more strongly on the last few digits as a neighborhood. Second, it shows that delayed generalization is not restricted to difficult global arithmetic tasks. Even this local rule can display a long optimization plateau followed by abrupt success. Third, it suggests that tiny chat-formatted transformers can learn stable symbolic shortcuts while retaining small but measurable failure modes.

The residual errors are also informative. Most mistakes are odd numbers predicted as even, and errors concentrate on final digits 7, 5, and 1, mostly at medium lengths. This asymmetry suggests that the learned rule is extremely robust but still slightly biased toward the positive class. The model is therefore better understood as a near-symbolic decision system than as an exact symbolic interpreter.

6 Limitations

This study has three main limitations. First, the task is intentionally simple and local, so conclusions should not be transferred directly to harder arithmetic or formal-language settings. Second, the attention analysis still stops short of full circuit tracing: while we now report head-level behavior and alternative normalizations, a fuller study could add activation patching, residual-stream decomposition, value-vector analysis, or head ablation. Third, our delayed-transition analysis relies on the saved training history and a confirmatory rerun, but not on checkpoint-by-checkpoint train-accuracy measurements, so we cannot make a stronger claim about canonical grokking.

7 Conclusion

We presented a compact empirical study of decimal parity learning in a tiny causal transformer. The model generalizes extremely well across lengths up to the context limit, learns a strongly rule-consistent dependence on the final digit, and exhibits a delayed plateau-then-jump training trajectory. At the same time, its internal behavior is not captured by layer-averaged last-digit attention alone: one head behaves like a partial last-digit pointer, while later layers emphasize the suffix as a region, and attention summaries remain looser than causal interventions. This makes the model a useful toy setting for studying the relationship between attention, causal importance, and delayed generalization in small transformers.

References

[1] Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization Beyond Overfitting on Small Algorithmic Datasets. arXiv:2201.02177, 2022.

[2] Hiroki Furuta, Gouki Minegishi, Yusuke Iwasawa, and Yutaka Matsuo. Towards Empirical Interpretation of Internal Circuits and Properties in Grokked Transformers on Modular Polynomials. arXiv:2402.16726, 2024.

[3] David Demitri Africa, Sara M. Kapoor, Theo Simon Sorg, and Challenger Mishra. Learning Modular Exponentiation with Transformers. arXiv:2506.23679, 2025.

[4] Alexander Kozachinskiy, Tomasz Steifer, and Przemyslaw Walega. Parity, Sensitivity, and Transformers. arXiv:2602.05896, 2026.

[5] Laura Gomezjurado Gonzalez. The Long Delay to Arithmetic Generalization: When Learned Representations Outrun Behavior. arXiv:2604.13082, 2026.

[6] Sarthak Jain and Byron C. Wallace. Attention is not Explanation. Proceedings of NAACL-HLT, 2019.