Validating Safety Guarantees of LSTM Models in MR Context

Kaiming Huang
kzh529@psu.edu
The Pennsylvania State University
State College, PA, USA

Peng Wu
wu.p@@northeastern.edu
Northeastern University
Boston, MA, USA

Mahdi Imani
m.imani@@northeastern.edu
Northeastern University
Boston, MA, USA

Tian Lan
tlan@gwu.edu
George Washington University
Washington, DC, USA

Gang Tan
gtan@psu.edu
Pennsylvania State University
University Park, PA, USA

Abstract

Ensuring the safety of neural network (NN) models in mixed reality (MR) systems is challenging due to adversarial manipulation of system parameters. We present PolySafe, which extends DeepPoly and Prover to validate safety of LSTM-based MR models. PolySafe unrolls temporal dependencies, introduces multi-plane abstractions for tighter bounds, and establishes probabilistic safety guarantees. It further includes an adaptive search that identifies minimal sets of critical parameters required to be constrained for defense. Evaluation on an MR engagement prediction model shows that PolySafe provides rigorous and actionable safety assurances for deployment.

CCS Concepts

• **Security and privacy → Logic and verification**.

ACM Reference Format:
Kaiming Huang, Peng Wu, Mahdi Imani, Tian Lan, and Gang Tan. 2025. Validating Safety Guarantees of LSTM Models in MR Context. In *The Twenty-sixth International Symposium on Theory, Algorithmic Foundations, and Protocol Design for Mobile Networks and Mobile Computing (MobiHoc '25), October 27–30, 2025, Houston, TX, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3704413.3765300

1 Introduction

Neural networks, such as LSTMs, are widely used in MR systems for tasks such as predicting user engagement. However, their predictions can be compromised by adversarial manipulation of system parameters (e.g., frame rate, latency), creating significant safety risks. Verifying that model outputs remain within specified safe limits under all relevant perturbations is therefore essential for deploying NN-based MR systems with confidence.

PolySafe extends DeepPoly [2] and Prover [1], which offer robustness guarantees for feed-forward and recurrent networks but lack mechanisms for probabilistic safety reasoning. PolySafe addresses this limitation by developing sound and tight probabilistic validation for LSTM-based MR models, delivering rigorous yet actionable guarantees, enabling practical defenses in safety-critical MR deployments.

MobiHoc '25, Houston, TX, USA

ACM ISBN 979-8-4007-1353-8/25/10
https://doi.org/10.1145/3704413.3765300

2 Formal Safety Guarantee

Our objective is to verify probabilistic safety guarantees of the form:

$$\Pr[L \le g(\mathbf{x}) \le U \mid x_{t,j} \in [\ell_j, u_j],\ j \in S] \ge \gamma, \tag{1}$$

where $g(\mathbf{x})$ denotes the LSTM model prediction, L and U are the safety bounds of the prediction, S indexes attacker-controllable parameters, and $[\ell_j, u_j]$ are the permissible ranges for each. Constraining $x_{t,j}$ within these intervals acts as a *defense metric*: it ensures under all allowed perturbations, the probability of maintaining safe outputs is at least γ. PolySafe quantifies the relationship between defensive parameter bounds and operational safety, allowing defenses to enforce and reason about risk-aware safety thresholds.

3 Extending DeepPoly for LSTM Verification

DeepPoly tracks for each neuron using both a simple interval capturing its minimum and maximum possible values, and an affine enclosure that relates the neuron to preceding activations. At each step, it propagates whichever bound is tighter. This approach is effective for feed-forward networks but assumes an acyclic graph; in LSTMs, feedback between hidden and cell states creates cycles, and single-plane relaxations (e.g., for $\sigma(u)\tanh(v)$) accumulate error across time steps, often producing vacuous bounds.

PolySafe improves LSTM validation by three improvements:

- **Temporal Unrolling:** PolySafe unfolds the LSTM for a fixed time horizon T, replicating all gates and states at each step:

$$(h_t, c_t) = \text{LSTMCell}(x_t, h_{t-1}, c_{t-1}), \quad t = 1, \ldots, T, \tag{2}$$

  producing an acyclic graph amenable to symbolic analysis.
- **Multi-Plane Relaxations:** For each nonlinear gate, e.g., $f(u,v) = \sigma(u)\tanh(v)$, PolySafe constructs K hyperplanes $\{L_k(u,v)\}_{k=1}^{K}$ for each input domain and forms a convex combination:

$$L_\lambda(u,v) = \sum_{k=1}^{K} \lambda_k L_k(u,v), \quad \lambda \in \Delta^{K-1}. \tag{3}$$

  This reduces over-approximation v.s. the single-plane method.
- **Adaptive Envelope Refinement:** After unsuccessful verification, PolySafe adapts convex weights λ via projected gradient ascent, where η is the verification margin. This tightens bounds

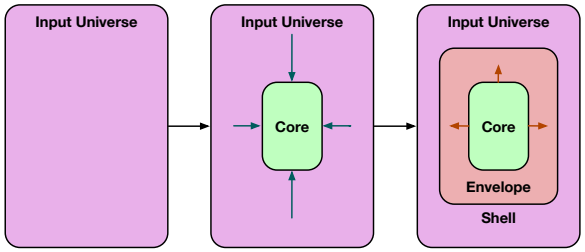

Figure 1: Core–Shell Safety Verification. Between deterministic core and shell is the envelope. The flow from left to right visually illustrates the iterative shrinking to a deterministic core, expansion to a shell, and identification of the envelope for probabilistic validation.

locally only where needed:

$$\lambda \leftarrow \Pi_{\Delta^{K-1}}(\lambda + \alpha \nabla_\lambda \eta), \tag{4}$$

Together, these enhancements allow PolySafe to derive sound, tight bounds for LSTM models, which was not achievable with DeepPoly or Prover's original single-plane approach.

4 Enhancing Prover for Probabilistic Guarantee

Prover extends DeepPoly by unrolling recurrent computation graphs, enabling robustness certificates for RNNs. However, it uses only single-plane abstractions, offers universal (worst-case) guarantees, and lacks mechanisms for probabilistic reasoning.

PolySafe addresses these gaps as follows (Figure 1):

(1) **Core Identification:** Iteratively shrink the parameter domain until a hyper-box C satisfies the safety property:

$$C = \prod_{j \in S} [\ell'_j, u'_j] \subseteq \prod_{j \in S} [\ell_j, u_j]. \tag{5}$$

(2) **Shell and Envelope:** Expand C to a shell region $\mathcal{S}$, with the envelope $\mathcal{E} = \mathcal{S} \setminus C$ capturing the gap between deterministic and probabilistic guarantees.

(3) **Probabilistic Guarantee:** Uniformly sample N configurations from $\mathcal{E}$. If K satisfy $L \le g(\mathbf{x}) \le U$, then the Clopper–Pearson exact bound below with confidence $1 - \alpha$.

$$\Pr(L \le g(\mathbf{x}) \le U \mid \mathbf{x} \in \mathcal{E}) \ge p_\ell, \tag{6}$$

Combining core and envelope, the shell guarantee is

$$\Pr(L \le g(\mathbf{x}) \le U \mid \mathbf{x} \in \mathcal{S}) \ge \rho + (1 - \rho) p_\ell, \tag{7}$$

where $\rho = \mathrm{vol}(C)/\mathrm{vol}(\mathcal{S})$.

This yields a probabilistic safety guarantee parameterized by user-specified confidence $1 - \alpha$, extending Prover's deterministic results to practical, risk-aware deployment.

5 Adaptive Minimal Subset Search

Given the probabilistic safety guarantee in Eq. (1), our objective is to identify the smallest subset $S^* \subseteq S$ of parameters such that

$$\Pr[L \le g(\mathbf{x}) \le U \mid x_{t,j} \in [\ell_j, u_j],\ j \in S^*] \ge \gamma. \tag{8}$$

PolySafe integrates this search into the core–shell framework. It begins by systematically enumerating candidate subsets S^* in order of increasing size, ensuring that simpler defenses are considered first. For each candidate, the intervals $[\ell_j, u_j]$ are adaptively refined. The refined subset is then evaluated using the core–shell procedure: if both the deterministic core and the probabilistically verified shell satisfy the guarantee, the subset is deemed sufficient. The search terminates once such a subset is found.

This integration yields a minimal S^* along with explicit interval bounds, ensuring that only the necessary parameters are constrained to achieve the target safety probability γ. In practice, this approach directly translate the constraints into actionable defenses.

Feature/Capability	DeepPoly	Prover	PolySafe
Feed-forward verification	✓	✓	✓
Recurrent/LSTM support	×	✓(loose)	✓
Multi-plane, adaptive refinements	×	×	✓
Probabilistic (core-shell) guarantees	×	×	✓
Minimal parameter subset identification	×	×	✓

Table 1: Comparison of PolySafe and previous approaches.

6 Case Study: MR Engagement Model

We evaluate PolySafe on an MR engagement prediction model: a two-layer LSTM that processes multiple one-minute sequences, each consisting of 40 features per time step (31 physiological and 9 system parameters such as FPS). The model is trained to output engagement scores from temporal inputs.

Using symbolic unrolling with multi-plane abstractions, PolySafe derives interval bounds and applies the core–shell procedure to establish probabilistic guarantees. Within this procedure, it performs a minimal-subset search, repeatedly invoking core–shell verification to determine the smallest set of system parameters that must be constrained to ensure safety. This yields clear operational guidance:

If FPS is constrained to remain within [100%, 120%] of its observed maximum in the input dataset, then with 90% confidence, the engagement score stays above the median value (level 4, for this model's 1–7 scale) for at least 99.99% of all possible scenarios.

Such results specify exactly which parameters require control, enabling reliable and cost-effective MR deployments.

7 Conclusion

In this paper, we presented PolySafe, a formal framework for soundly validating the safety of LSTM-based models in MR. By extending DeepPoly and Prover with temporal unrolling, adaptive multi-plane abstractions, core–shell probabilistic verification, and minimal parameter subset search, PolySafe delivers tight probabilistic safety guarantees. This approach identifies the minimal system controls required for safe MR deployment, providing a foundation for the reliable use of complex neural network models.

Acknowledgments

This work was supported in part by the Defense Advanced Research Projects Agency (DARPA) under grant HR0011-2420366.

References

[1] Wonryong Ryou, Jiayu Chen, Mislav Balunovic, Gagandeep Singh, Andrei Dan, and Martin Vechev. 2021. Scalable Polyhedral Verification of Recurrent Neural Networks. In *Computer Aided Verification: 33rd International Conference, CAV 2021*. Springer-Verlag, Berlin, Heidelberg, 225–248.

[2] Gagandeep Singh, Timon Gehr, Markus Püschel, and Martin Vechev. 2019. An abstract domain for certifying neural networks. *Proc. ACM Program. Lang.* 3, POPL, Article 41 (Jan. 2019), 30 pages.