On the Role of Formal Languages as Primary Training Media for Foundation Models: Efficiency, Reasoning, and Semantic Generalization

Thomas Scheuerle
Technical Working Group for Formal Automotive Systems (TWG·FAS)

August 4, 2026

Abstract

Recent advances in foundation models have been driven predominantly by large-scale pretraining on natural language corpora. While natural language provides broad world knowledge and diverse semantic structures, it also introduces substantial ambiguity, redundancy, and syntactic variability. This paper examines the hypothesis that training a model predominantly on a formal language can significantly reduce model complexity, training cost, and inference cost for specialized domains. At the same time, such specialization may inhibit the emergence of certain higher-level capabilities associated with large-scale natural language training. We discuss the theoretical foundations of this trade-off, propose methods for quantifying the expected efficiency gains, identify risks related to capability loss, and outline design principles for future formal-language-centric learning systems. Particular attention is given to formal specification languages for safety-critical cyber-physical systems and autonomous vehicles.

1 Introduction

Natural language has become the universal interface for modern foundation models. Its success derives from the enormous amount of available training data and the broad range of human knowledge encoded within it. However, natural language possesses characteristics that are not necessarily desirable for engineering tasks:

- ambiguity,
- synonymy,
- context dependence,
- implicit assumptions,
- linguistic redundancy.

In contrast, formal languages are intentionally designed to minimize ambiguity and maximize semantic precision. Examples include programming languages, temporal logics, specification languages, theorem-proving languages, and domain-specific languages (DSLs). This raises a fundamental question:

> Can foundation models be made substantially smaller, faster, and more reliable by training predominantly on a well-designed formal language?

The answer appears to be affirmative for certain classes of tasks. However, the resulting gains may come at the cost of reduced capability emergence, weaker contextual reasoning, and diminished semantic flexibility.

2 The Semantic Compression Hypothesis

A central observation is that formal languages act as semantic compressors. Consider the following requirement:

> "The vehicle shall initiate braking within one second whenever a pedestrian enters the collision zone."

A corresponding formal representation may be

$$G(\texttt{pedestrian\_conflict} \rightarrow F_{[0,1s]}(\texttt{brake}))$$

or an equivalent domain-specific notation. Although both expressions convey nearly identical semantics, the formal representation removes:

- paraphrases,
- stylistic variations,
- grammatical complexity,
- linguistic uncertainty.

The model therefore allocates less capacity to understanding alternative formulations and more capacity to capturing semantic structure. We refer to this phenomenon as *semantic compression.*

3 Expected Advantages

3.1 Reduced Training Complexity

Formal languages typically exhibit:

- smaller vocabularies,
- constrained grammars,
- fewer valid token transitions,
- lower entropy per semantic statement.

Consequently fewer parameters may be required to reach a given accuracy level.

3.2 Improved Sample Efficiency

Because semantic distinctions are represented explicitly, each training example tends to provide denser supervision. The model spends less effort learning linguistic equivalence classes and more effort learning domain semantics.

3.3 Faster Inference

Formal languages enable:

- grammar-constrained decoding,
- type checking,
- parser-guided generation,
- validity-preserving search spaces.

The reduction of syntactic uncertainty can improve decoding efficiency and output reliability.

3.4 Improved Verifiability

Formal outputs can typically be:

- parsed,
- type checked,
- simulated,
- formally verified,
- transformed into executable monitors.

This provides objective correctness criteria unavailable in ordinary language generation.

4 Potential Disadvantages

4.1 Loss of Semantic Diversity

Natural language contains far more than syntax. It encodes:

- causal narratives,
- analogies,
- explanations,
- commonsense knowledge,
- social interactions,
- planning strategies.

Removing such material risks reducing the diversity of reasoning patterns encountered during training.

4.2 Reduced Emergent Capabilities

Many remarkable properties of large foundation models appear to arise from exposure to highly diverse data distributions. Possible losses include:

- transfer learning,
- abstraction,
- analogy formation,
- open-ended problem solving,
- contextual adaptation.

A model trained exclusively on a narrow formal language may become highly specialized while losing broader reasoning abilities.

4.3 World Model Degradation

Formal languages describe the world but are not themselves the world. A sufficiently compressed representation may omit information that humans routinely infer from natural language. The danger is the creation of a system that manipulates symbols correctly while possessing an impoverished understanding of the phenomena represented by those symbols.

4.4 Brittleness

Formal languages frequently tolerate little deviation. A single symbol may invert the semantics of an entire expression. For example,

$$G(p \to q)$$

and

$$G(p \to \neg q)$$

differ by one token but may represent opposite safety requirements. A model must therefore develop considerably stronger semantic precision.

5 Quantifying Complexity Reduction

A central question is how much reduction can be expected. Consider the training loss

$$L(N, D) = E + \frac{A}{N^\alpha} + \frac{B}{D^\beta}$$

where

- N denotes model parameters,
- D denotes training data,
- E denotes irreducible error.

For a target performance level ϵ one may estimate

$$N_{NL}(\epsilon)$$

for natural language training and

$$N_{FL}(\epsilon)$$

for formal-language training. The parameter reduction factor becomes

$$R_N = \frac{N_{NL}(\epsilon)}{N_{FL}(\epsilon)}.$$

Similarly, one may define corresponding data and compute reduction factors. This provides an experimentally measurable quantity rather than a purely qualitative claim.

6 Dependence on Language Design

Not all formal languages are equally suitable. The risk of capability loss depends strongly on language properties. Desirable properties include:

- compositional semantics,
- strong typing,
- explicit state representation,
- executable semantics,
- canonical syntax.

Undesirable properties include:

- excessive compression,
- overloaded operators,
- hidden assumptions,
- semantic incompleteness.

A successful formal language must dramatically reduce ambiguity without eliminating meaningful semantic structure.

7 Mitigation Strategies

The most promising approach may not be exclusively formal-language training but rather formal-language-dominant training. Several strategies appear particularly attractive.

7.1 Parallel Representations

Train on aligned triples:

1. natural-language description,
2. formal specification,
3. executable semantic interpretation.

7.2 Semantic Grounding

Connect every formal statement to:

- simulations,
- traces,
- state transitions,
- counterexamples.

7.3 Reasoning Traces

Provide intermediate reasoning artifacts such as:

- proof sketches,
- dependency graphs,
- state-transition chains,
- scenario decompositions.

7.4 Counterexample-Guided Learning

Integrate formal verification tools directly into the training loop. Incorrect specifications should produce explicit counterexamples that become additional training signals.

8 Implications for Autonomous Systems

Autonomous vehicles provide an especially compelling application domain. Current engineering processes often require translation between:

- stakeholder requirements,
- natural-language specifications,
- test descriptions,
- implementation artifacts.

Each translation step introduces ambiguity. A formal specification language serving as a primary representation layer could reduce ambiguity while supporting automated verification and machine-assisted engineering. This suggests a future architecture in which natural language is used primarily as a human interface while formal languages become the dominant medium for machine reasoning and verification.

9 Conclusion

Training foundation models predominantly on a formal language offers a powerful path toward smaller, faster, and more verifiable systems. The resulting models may require significantly fewer parameters and less training data for tasks that are naturally expressible within the language's semantic domain. However, the advantages are accompanied by substantial risks. Natural language serves not merely as a communication medium but also as a carrier of diverse reasoning patterns and world knowledge. Excessive specialization may therefore inhibit capability emergence and semantic generalization. The central challenge is not whether formal languages should replace natural language entirely. Rather, it is the design of hybrid training ecosystems that combine the precision of formal representations with the richness and diversity of natural-language-derived semantic structures. Future research should focus on experimentally quantifying the trade-off between semantic compression and capability retention and on designing formal languages that maximize both efficiency and generalization.