Beyond the Prompt: Assessing Domain Knowledge Strategies for High-Dimensional LLM Optimization in Software Engineering

Srinath Srinivasan, Tim Menzies

Computer Science, **North Carolina State University**
ssrini27@ncsu.edu
timm@ieee.org

MSR 2026 — Registered Reports Track
13 April 2026, Rio de Janeiro, Brazil





Research Goal & The Dimensional Barrier

The goal of this research is to find a way to overcome LLM generated warm-starts for SE optimization in high-dimensional problems.

Context: The Dimensional Barrier

- Ground truth data for SE optimization is prohibitively expensive [Valov17]
- LLMs can generate cheap "warm starts", but only for simple problems
- **The Collapse**: LLMs succeed at low dimensions (<=5) but completely fail at high dimensions (>11) [Senthil24]
- **The Cause**: Lack of public training data for niche, complex SE configurations [Villalobos24]

Approach: The Registered Report

- Evaluate four distinct architectures to inject missing domain knowledge
- Compare **Human-guided** vs. **Fully Automated** reasoning
- Test against authoritative Bayesian models (Gaussian Process)

Strategies (1/2):

Method 1: H-DKP (Human-in-the-Loop)

- **Motivation**: Public documentation misses the “unwritten rules” of a system.
- **Intuition**: Experts can easily spot why an LLM failed, even if they cannot write the perfect prompt from scratch.
- **Method**: 10-day asynchronous loop. LLM fails -> Expert answers “Why?” via email -> Knowledge base updates automatically.

Method 2: AMP (Adaptive Multi-Stage Prompting)

- **Motivation**: Asking an LLM for an immediate answer causes logical hallucinations.
- **Intuition**: Forcing the model to explicitly write out constraints first improves its own generation.
- **Method**: 4 sequential stages (Analyze, Discover Constraints, Generate, Self-Validate). Tests if LLMs can police their own boundaries.

Strategies (2/2):

Method 3: DAPR (Dimension-Aware Progressive Refinement)

- **Motivation**: Balancing 30+ features simultaneously overwhelms LLM attention.
- **Intuition**: Hide the complexity. Secure the most critical variables first, then fine-tune the noise.
- **Method**: Statistically rank features [Pfisterer22]. Prompt the LLM to optimize only the top few. Lock those in, and progressively expand the feature space.

Method 4: HKMA (Hybrid Knowledge-Model Approach)

- **Motivation**: LLMs understand semantics but fail at math. Bayesian models excel at math but lack context.
- **Intuition**: Ground the LLM's language skills in empirical, statistical reality.
- **Method**: Run a fast statistical scout (TPE) to find data priors [Bergstra11]. Use RAG to fetch docs explaining those priors [Lewis20]. Prompt LLM to synthesize using both.

Evaluation Plan: The MOOT Testbed

How we will prove this works without breaking the bank.

The Dataset (MOOT Repository):

- 120+ software engineering optimization tasks
- Stratified exactly by complexity: Low (<6), Medium (6-11), High (>11)
- Broad domains: Cloud configurations, Scrum models, Project Health

Primary Metrics:

- **Quality**: Chebyshev Distance (how close is the generated start to the theoretical optimum?)
- **Efficiency**: Total API tokens consumed per successful generation.

Statistical Rigor:

- **Scott-Knott Clustering**: Generates strict, non-overlapping performance ranks [Scott74].
- **Cliff's Delta**: Ensures we only report differences with practical, real-world effect sizes [Cliff93].

Selected References

[Ahmed25]: Ahmed, T. et al., "LLMs function best as assistive rather than authoritative agents," *MSR*, 2025.

[Bergstra11]: Bergstra, J. et al., "Algorithms for hyper-parameter optimization," in *NIPS*, 2011.

[Cliff93]: Cliff, N., "Dominance statistics," *Psychological Bulletin*, 1993.

[Krishna20]: Krishna, R. et al., "Whence to learn? Transferring knowledge," *TSE*, 2020.

[Lewis20]: Lewis, P. et al., "Retrieval-augmented generation for knowledge-intensive NLP," *NeurIPS*, 2020.

[Nair17]: Nair, V. et al., "Using bad learners to find good configurations," *FSE*, 2017.

[Pfisterer22]: Pfisterer, F. et al., "Surrogate NAS benchmarks," *PMLR*, 2022.

[Scott74]: Scott, A. J. et al., "A cluster analysis method for grouping means," *Biometrics*, 1974.

[Senthil24]: Senthilkumar, M. et al., "LLMs improve active learning warm starts," *MSR*, 2024.

[Villalobos24]: Villalobos, P. et al., "Will we run out of data?" *ArXiv*, 2024.