

Tholos-2B: Technical report

Mert Kaya

Eschatia Labs, 2026, eschatialabs.com

Technical report (not peer-reviewed). Every figure comes from the published sources listed at the end.

What it is

Tholos-2B is MiniCPM5-2B fine-tuned to be the agent in Tholos, an app where a small team of agents shares tables, notes and a task board on your own machine. On each turn the model writes one JSON step: a short thought, the name of a tool and that tool’s arguments. The weights and the GGUF builds are released under Apache-2.0.

Method

The model was trained with LoRA, using Unsloth, and the adapter was merged into the released weights. The loss covers the assistant’s JSON step; system, user and tool messages are masked.

The training data is synthetic. Teacher models played the agent inside the Tholos runtime, and a trajectory was kept only if the final state of the workspace passed the scenario’s checks. The data covers ordinary workspace tasks and prompt-injection scenarios, where a table, a note or a fetched page carries instructions the agent should ignore.

Tholos-Bench ships with Tholos and has 160 scenarios, each scored on the final state of a workspace.

Results on Tholos-Bench

Tholos-2B passes 137 of 160 scenarios, 25 more than MiniCPM5-2B, the model it was trained from. The table sets it beside models of its own size and larger ones, all as Q4_K_M files on one Kaggle T4 with llama.cpp and JSON schema decoding. Only Tholos-2B was fine-tuned for the Tholos step format.

Model, Q4_K_M	File	Passed, of 160	Injection, of 12
Qwen3.5-2B	1.28 GB	97	4
MiniCPM5-2B (base of Tholos-2B)	1.56 GB	112	8
Tholos-2B	1.56 GB	137	10
Llama 3.2 3B Instruct	2.02 GB	37	1
Granite 4.2-3B	2.24 GB	129	7
Qwen3.5-4B	2.74 GB	141	6

Three ways to measure

One file scores differently on each inference path, so compare scores inside one path. CPU llama.cpp with the JSON schema is the reproducible path, Ollama is the default for most users, and the Kaggle T4 run is the comparison across models above.

Path	Server and mode	MiniCPM5-2B	Tholos-2B
CPU llama.cpp	llama-server b11263, 4 threads, JSON schema	117	134
CPU Ollama	Ollama 0.32.4, 4 threads, the repo's template and params, JSON mode	96	136
Kaggle T4	llama.cpp CUDA build fdf5818, JSON schema	112	137

Limits

- The training data and the benchmark are in English, and evaluation covers Tholos-Bench only. Behavior on other tasks, tool sets and prompt formats is unmeasured.
- Categories are small, with 12 scenarios for injection, so a gap of one or two scenarios between models is within noise.
- The data comes from teacher models and was kept by lexical checks, so the model inherits the teachers' habits and the checks' blind spots.
- Run it with schema-constrained decoding or validate each step before anything acts on it. Keep a person in the loop for actions with real consequences, and keep approvals on for any action that fetched text could misuse.

Sources

- [Tholos-2B model card](#)
- [Tholos-2B GGUF builds](#)
- [Tholos source code and Tholos-Bench](#)
- [Published training trajectories](#)

Citation

```
@techreport{kaya2026tholosreport,  
  author = {Kaya, Mert},  
  title = {Tholos-2B: Technical report},  
  institution = {Eschatia Labs},  
  year = {2026},  
  url = {https://eschatialabs.com/research/tholos-2b-report/},  
  note = {Technical report, not peer-reviewed}  
}
```