

Deterministic intelligence at the edge.

Structuring chaotic developer signals into strictly valid JSON on one CPU core. Any text in, typed records out - measured, then published.

MODEL mivia-small-json	CHAIN mm12-realworld-r1-s42	HARDWARE cpu only - no gpu
SIZE ON DISK ~650 mb int8 artifacts	SURFACES rest + streamable-http mcp	MEMORY < 2 gb - models load once

A generative model is the wrong tool for a deterministic task.

	CLOUD GENERATIVE LLM	MIVIA-SMALL-JSON
latency	716 ms p50 (network)	~3.3 ms p95 (full local path)
format errors	0 (same 350-row set)	0
accuracy	97.14%	99.71%
cost	\$0.0081 / 350 rows	\$0 marginal self-host - \$0.000005 hosted
data	leaves the box	never leaves the box

Same 350 test prompts on both sides, no tuning on either side. The hosted model is far larger; on workflow structuring a ~650 MB artifact matches or beats it at ~60x the speed.

Small enough to live where the text already is.

PIPELINE LATENCY route + extract ~3.3 ms p95	PIPELINE LATENCY < 12 ms p95	DUMP LATENCY ~11k chars in 25-87 ms
ARTIFACTS int8 - sha256-verified - no pickle	DEPLOYMENT single non-root container	AUTH bearer + host allow-list + per-key metering
NETWORK no outbound calls at inference	RETENTION processed in memory, discarded	IMAGE one docker image, cpu-only

Twelve record types. Every input gets one that fits.

tool_call	agent calls	diff	code & vcs
tool_clarification	agent calls	review_comment	code & vcs
no_tool_call	agent calls	log_event	ops
issue	code & vcs	diagnostic	ops
commit	code & vcs	agent_step	ops
pull_request	code & vcs	text	general - first-class, never forced

Non-workflow text is a first-class outcome: 92.2% of 605 non-workflow rows classified as text, zero spurious predictions on 350 in-domain rows.

Every tool request gets a verdict before anything runs.

CALL

Match with complete arguments.
Executes - deterministic veto
already passed.

CLARIFY

Match with missing arguments.
Returns a structured question
naming the parameters - never a
guessed call.

ABSTAIN

No matching tool. Declines instead
of forcing; falls back to
conversation.

"delete the database" can never execute with empty arguments. 95.4% of underspecified requests handled without a guessed call; guard abstains on 79.7% of 5,100 no-tool rows while retaining 95.1% of real calls.

What it never does, written down.

INPUT	A GUESSING MODEL EMITS	THIS ONE DOES
"GET /health 200 in 4ms"	"ts": "2025-01-01T..."	ts absent - never fabricated
"commit abc12"	"sha": "abc1200...0"	"sha": "abc12" - never padded
"delete the database"	call delete({})	clarify: "which database?"
binary garbage	200 { ... }	422 - honest 4xx, never a wrong 200
11k-char dump	truncated record	complete - bounded only by request size

Your candidate list, ranked. An honest no-decision when nothing fits.

```
in {"text": "users in the eu report checkout fails during 3d secure...",  
    "candidates": ["Authentication & Accounts", "Payments & Checkout",  
                  "Fraud & Risk", "Network & Infrastructure"]}  
out ("Authentication & Accounts", 0.62, "abstain") // no decision
```

CALIBRATED STATUS

"ok" picks, "ambiguous" and
"abstain" decline - declined rows
go to rules or a human, not a
forced pick.

MEASURED GATING

Conformal gating (alpha 0.12):
83.8% of 77-intent rows called at
88.3% called-accuracy.

OPEN SET

Banking77 zero-shot: 81.0% top-1
where chance is 1.3%; trained
intent head: 83.2%.

Four tiers, your desk's labels. Within-one is the operative read.

SURFACE	EXACT	WITHIN-ONE
register panel (n=217)	70.0%	93.1%
real business tickets, production (n=600)	46.8%	84.2% QWK 0.675
german tickets (370-row held-out)	43.0%	90.5%

WHY WITHIN-ONE

Severity is subjective: human annotators agree on the exact level only ~72% of the time. The crosswalk renders tiers into PagerDuty SEV, Jira P-levels and Bugzilla severities.

THE CALL

grade({text}) - prose or a pasted record alike. The tier is the model's call; the vocabulary is yours.

Sixteen examples. Zero retraining.

FEW-SHOT ONBOARDING

92.98% record-exact (mean) across five unseen domains at 16 examples per class.

ENUM ALIASES

urgent | eilt | dringend | فوری |
紧急 | 긴급 → "high" - the declared
vocabulary is yours, the tier is
the model's.

THE HONEST SPLIT

Intent-style schemas measure in the
90s; verbatim-span schemas reach
33.3% record-exact. Both regimes
are quoted as measured.

A per-schema extractor trains on-CPU in seconds. The schema declaration is the whole integration - the model bytes never retrain.

Held-out, then published. Every number names its set.

SURFACE	SET	RESULT
routed + valid	350 held-out rows	99.71% (rules baseline 79.71%)
json validity	350 held-out rows	100%
loghub 16 systems	32,000 lines	100% routed / 100% valid
swe-bench → issue	100 statements	99%
bfcl tool call	183 public rows	88.0% (rest abstain)
german tickets → issue	300 rows	98.7%
surface retention	all held-out	1.0 (tickets-de 0.99)

What it cannot do, shown as-is.

SURFACE	RESULT	READ
JevBench v1 general decisions	54.1% (easy 85.4 / original 52.8 / hard 41.4)	open-domain, never tuned for
S1MB: 106 task families	35.5% (26,269 decisions)	out-of-domain, shown as-is
severity exact-match	46.8% (600 real tickets)	subjective - humans ~72%
diffs: rules baseline	98% routed vs rules 100%	the one row the baseline wins
pytest-family lines	localize weakly	pair records with raw lines

An abstain is a feature, not a failure - declined rows go to rules or a human. Forced picks are what this model exists to prevent.

Five methods. One bearer key. One billing surface.

<code>POST /v1/models/mivia-small-json/extract</code>	<code>{"text": "..."}"</code>	→ typed records
<code>POST /v1/models/mivia-small-json/structure</code>	<code>{"text", "schema"}"</code>	→ record per declared schema
<code>POST /v1/models/mivia-small-json/select</code>	<code>{"text", "candidates"}"</code>	→ (choice, score, status)
<code>POST /v1/models/mivia-small-json/grade</code>	<code>{"text": "..."}"</code>	→ action + four-tier
<code>POST /v1/models/mivia-small-json/guard</code>	<code>{"request", "tools"}"</code>	→ CALL / CLARIFY / ABSTAIN

HOSTED

\$5 / 1M calls - \$0.000005 each. First 1,000 calls every month are free; prepaid balance after. 429 on empty, never an overage bill.

SELF-HOSTED

The full surface at the container root (/v1/route, /v1/repair, /v1/extract ...), same image, your hardware - zero marginal cost.

Measured, then published. Nothing else is claimed.

Every number above names its set and comes from frozen, never-trained-on evaluation on CPU. The model card carries the full tables, the guarantees, and the wire contract.

MODEL CARD /models/mivia-small-json	API DOCS /docs/api	SOURCE github.com/MiviaLabs/mivia-small-json
--	---------------------------------------	---