

LexiNexiAI

Small language models that keep learning

ANUPNET: label-free continual learning without stored data — built and tested by a one-person lab in Ujjain, India.

Founded Apr 2024 · Pvt Ltd · DPIIT

CIN U62011MP2024PTC070875

Founder

Anup Kumar Baghel

Contact

admin@lexinexiai.com

An AI product company that pays its own way

Live products (all built in-house), plus software services that fund the research.

1

WhatsApp AI Agent

Books consultations, answers customers 24/7. Live at +91 92443 48216.

2

Developer Agent (iOS)

AI coding app, approved on the Apple App Store.

3

ExaProcess

Billing software serving contractors since 2015.

4

AnimeMasti

Full e-commerce store: orders, billing, delivery.

Founder: 12 years building software at Johnson Controls, R Systems and ACL Digital. Serverless-first engineering; all deployments run on Google Cloud today.

The problem: language models stop learning the day training ends



Frozen weights

Every LLM today is a snapshot. New knowledge means a new training run.



Catastrophic forgetting

Fine-tune on a new domain and the old one is overwritten — in our tests, +2.74 nats of loss on the old domain.



Replay needs the past

The standard fix stores old data and mixes it back in. That fails when data is private, gone, or unbounded.



Cost lands on retraining

Enterprises re-train or re-index every few months instead of learning continuously.

ANUPNET: a hall, rooms, dreams and sleep

HALL — frozen trunk (skills, grammar, gist)

Room 1

gate wiki

Room 2

gate code

Room 3

gate stories

gates open only when their domain is recognised

DREAMS + SLEEP — model-generated text trains the gates, never the weights

1

Novelty

Loss jumps above the model's own noise band → open a new room before the step.

2

Familiarity

An old room's gate wakes up on the input → reopen that room instead.

3

Sleep

Silence rooms on dreams; refresh keys of older rooms; merge least-used rooms.

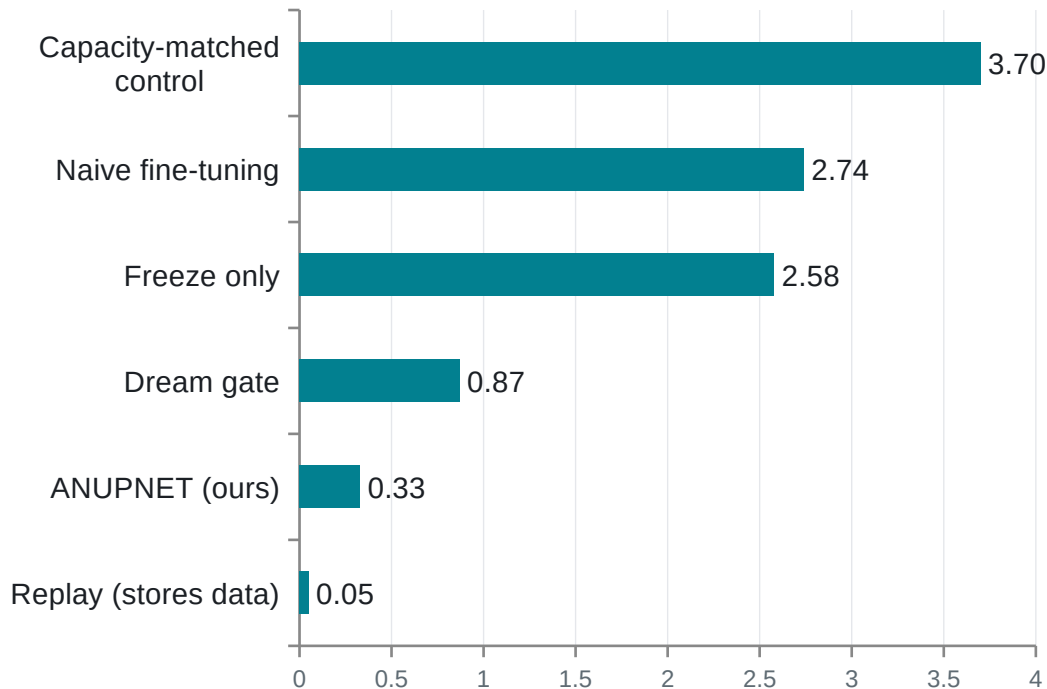
4

No absolute numbers

Every decision threshold is a ratio to the model's own statistics.

88% less forgetting — no stored data, no labels

30M-parameter model, wiki → code → stories. Average forgetting on old domains (nats, lower is better).



0.33 ± 0.05

3 seeds, vs 2.74 naive

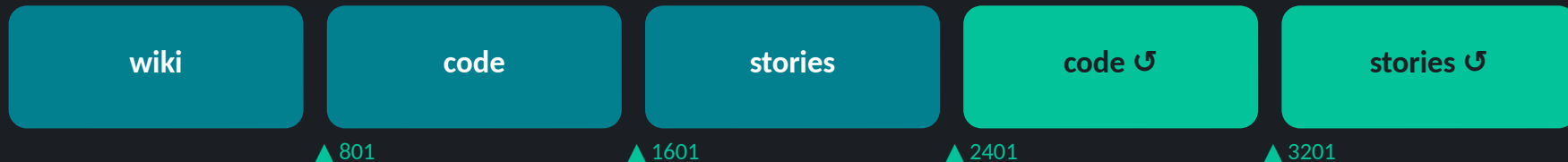
3.70

same extra params, no mechanism: worse than naive

~0.45 nats

acquisition cost on new domains — reported, not hidden

One unlabeled stream — the model finds the boundaries itself



▲ = boundary detected at the exact step (801, 1601, 2401, 3201) · ↀ = old room recognised and reopened

4 / 4

boundaries found, exact step

0

false alarms (3 seeds)

2 / 2

rooms opened = rooms needed



2nd visit better than 1st:
code 5.00→4.76, stories
3.72→3.46

Stage E: frozen Qwen 2B/9B + rooms + slot memory

Same three rules on top of a frozen open-weight model: the base is the hall (never trained), rooms learn skills, a hippocampus-style slot memory holds each fact. Daily stream, one sleep per night, old rooms offloaded to CPU and woken on demand.

Setting	Plain fine-tune	ANUPNET
9B hall, 6-day stream, retained facts / skills	—	13/14 • 19/20
2B hall, 10-day stream, learned facts	28% (template collapse)	60% (slot memory v3)
2B hall, 20-day stream, learned facts	22%	in progress (v4)
Cross-room forgetting after new learning	present	0 measured
Offload old room to CPU, recall on question	—	lossless (12/14, 17/20)

Single-seed so far; 30-day and 9B-scale confirmation runs are next and need sustained GPU. Runs on Google Cloud Run GPU jobs (NVIDIA RTX Pro 6000); 2B/9B = Qwen open-weight models.

Where we are: from 30M to a frozen 9B hall

DONE

A • Pretrain

110M params, 1.05B tokens (FineWeb-Edu + Cosmopedia), val loss 3.30, 6 GPU-hours on one RTX Pro 6000.

DONE

B • Instruction tune

SmolTalk, Q&A format; the model answers questions in English.

DONE

C • Frozen open-weight hall

Switched to Qwen 2B/9B as the hall; rooms + slot memory on top (Stage E).

IN PROGRESS

D • Continual stream

10–30 day streams on Qwen 2B/9B; 30-day + 9B multi-seed confirmation next; then product pilot: Agent Lexi that remembers each customer.

What we need from a compute partner

1

Compute credits

~200–500 H100-hours over 3–4 months for 30-day streams, 9B confirmation and 3 seeds. Today: self-funded on one cloud GPU; credits ending.

2

Technical guidance

PyTorch/CUDA performance and multi-GPU scaling of rooms + slot memory on 9B-class models.

3

Visibility

Co-marketing when the technical report and open-source code ship (arXiv + GitHub, Q1 2027).

What the partner gets: an open, reproducible continual-learning method (code + technical report, arXiv/ICLR-workshop 2027) developed end-to-end on NVIDIA GPUs, with a published cost comparison against replay and retraining, named as compute partner — and a product company that ships.

Models that learn every day, not every quarter.

Anup Kumar Baghel • Founder, LexiNexiAI Private Limited

admin@lexinexiai.com • +91 91797 08436 • lexinexiai.com • linkedin.com/in/lexinexiai

Technical report v0 and code available on request; public release with v1.