Sparks of Self-Healing:

Early experiments with CA3

AUTHOR
Alberto Acedo

AFFILIATION
Biome Makers Inc.

PUBLISHED
November 2026

Every number in this post is read from the run's records or taken from [1]. Independent work by Biome Makers Inc., not produced or endorsed by the authors of [1], MIT, OpenAI or Alibaba.

TL;DR

We find that a language model edited by another AI, with CA3 as its controller, can find what the edit broke by checking itself against its pre-edit self, and heal it from its own correct answers, with no outside teacher and without giving back what the edit gained. The model supplies the material: its own past and its own correct answers. CA3, our system for governing what a deployed model learns, is the controller: it runs the check, chooses what to heal and keeps the healing where it belongs. We call this self-healing.

Last week, GPT-6 Astra edited 0.016% of the weights of Qwen3.5-0.8B to fix its reasoning loops, and raised its MATH-500 score from 7.9% to 48.1% [1]. With CA3 as its controller, we asked the edited model to check itself. Nobody told it where to look, and it flagged one skill out of eight: asked to write a word a given number of times, it now often comes up one short.

It healed that skill from its own correct answers in 23 minutes on one GPU, from 76.3% to 98.5%, past the 92.0% of the model before the edit. The seven other skills kept exactly the same scores, and none of 1,619 math problems reaches the healing, so the edit's gain stays whole.

Every change to a model is chosen for what it improves. As models start to edit models, we think every change will also need a check-up of what else it changed, and a healing that touches nothing else.

FIND

Did the edit break anything?

Repetition

flagged: 1 of 8 skills, 92.7% → 84.3%

The edited model checks itself against its pre-edit self, with CA3. Nobody tells it where to look.

HEAL

Heal it from your own answers.

98.5%

up from 76.3%; before the edit, 92.0%

No outside teacher and nothing from the original, in 23 minutes on one GPU.

KEEP

Keep everything else.

0 of 1,619

math problems reach the healing

The seven other skills keep exactly the same scores, and the edit's gain stays whole.

If an AI edited 0.016% of your model's weights and its math score went up sixfold, would you ship it? And if the same edit had broken something no math benchmark looks at, how would you find out? How would you fix it without giving the gain back?

Today, a change to a model is judged by the benchmark it was made for. Problems in a model are found only if someone has a test designed for them, or worse, after deployment [1]. In this post, we find that the changed model itself, with CA3 as its controller, can answer both questions: it finds what the change broke by comparing itself with its own past, and heals it from its own correct answers while the gain stays.

Self-healing

Models change after they ship. They are fine-tuned on feedback, merged, compressed for devices and replaced by new versions, and each of these changes can break things the old model did right [2, 3, 4, 5]. Last week, Weight-Space Programming added a new way to change a model: a frontier model that reads a network's weights and edits them directly, with no gradients and no training [1]. The reaction went from "Forget retraining" in the press [6] to a step towards models that improve models [7].

Every one of these changes is chosen for a gain and measured by that gain. In self-healing, a model that has been changed checks itself against its own past, finds what the change broke, and heals it from its own correct answers, under a control that keeps the healing where it belongs. The model does not do this alone. It supplies the material: its own past and its own correct answers. What makes the healing happen is CA3 as the controller: it runs the check, chooses what to heal and controls what is written, so that nothing else breaks. Without such a control, a model that trains on its own output can raise its own metric and then collapse [8]. A model can improve itself on what it can check, but it cannot see what it lost without something fixed to compare against. That is why the check is made against its own past, kept unchanged.

With CA3, it becomes possible to just ask a changed model what the change broke, and then to ask it to heal that and touch nothing else. In this post, we share very early results on one change, the edit behind the sixfold MATH-500 gain of [1]: one small model, one edit and one run, pre-registered before anything was measured, with the code reviewed independently before it ran.

- A changed model can find what the change broke. In Experiment 1, the edited model, checking itself against its pre-edit self with CA3, flags one skill out of eight, the same one our earlier outside audit had found [9].
- It can heal it from its own answers. In Experiment 2, it heals that skill past where it started, with no outside teacher and nothing taken from the original.
- The healing can stay where it belongs. In Experiment 3, CA3 keeps the healing away from everything else, including every math problem behind the edit's gain.

What else did the edit change?

The edit cures a common failure of tiny reasoning models: they loop, repeating the same text until they run out of tokens. Astra traced the looping in Qwen3.5-0.8B to one direction written by the MLPs of layers 15 and 19, and subtracted it from 131,072 weights, 0.016% of the model. The share of MATH-500 answers that ran into the token limit fell from 93% to 33%, and MATH-500 rose from 7.9% to 48.1% [1]. Because such an edit touches so few weights, its authors write, it "can leave the rest of the model intact" [1].

We checked. With thinking off, the way Qwen3.5-0.8B runs by default [10], seven of the eight skills we measured stayed within two points of the original (Experiment 1). One plain skill, which no math benchmark looks at, broke. Ask the model to write a word a given number of times, and it often comes up short:

> Write the word "teacher" exactly 4 times, separated by commas, and nothing else.
>
> Original: `teacher,teacher,teacher,teacher` (4, right)
>
> Edited: `teacher, teacher, teacher` (3, one short)

Of the 100 answers the edit broke among our 600 judgement questions, 87 are exactly this: one word short. Interestingly, what broke is itself a kind of repetition, the kind you ask for. It is tempting to connect the two, an edit that removes a direction tied to looping and a model that now stops one word early, but we have not tested that link.

Our earlier outside audit of this edit had flagged the same skill [9]. The question was whether the model could find it without being told, and heal it without losing what the edit gained.

Can an edited model find what its edit broke?

EXPERIMENT 1

Can the edited model tell, by itself, what the edit broke?

We gave CA3 the original and the edited model and one exam of eight skills: copying a text exactly, repeating a word N times, returning JSON with given keys, refusing harmful requests and answering harmless ones [11], and three that were only watched: instruction following [12], a Spanish exam [13] and grade-school math [14]. Both models answered the same 1,150 questions of the revision half, and CA3 compared them question by question. We asked: did the edit break anything?

SETUP

the original model, the edited model and one exam of eight skills (1,150 revision questions)

Did the edit break anything?

Nobody says which skill to look at. A rule written before the run decides what counts as broken: 3 points or more down, with the 95% interval entirely below zero.

ANSWER

Broken: repetition

92.7% → 84.3% · -8.3 points (95% interval -12.7 to -4.0) · 1 of 8 skills · ✓ the skill our earlier outside audit had found [9]

One skill broke: repetition, from 92.7% to 84.3%; the edit lost 35 of those questions and gained 10. The seven others moved by two points or less, and none of them met the rule. This is exciting because nobody told CA3 where to look: the model's own past was enough to name the skill (Figures 1 and 2).

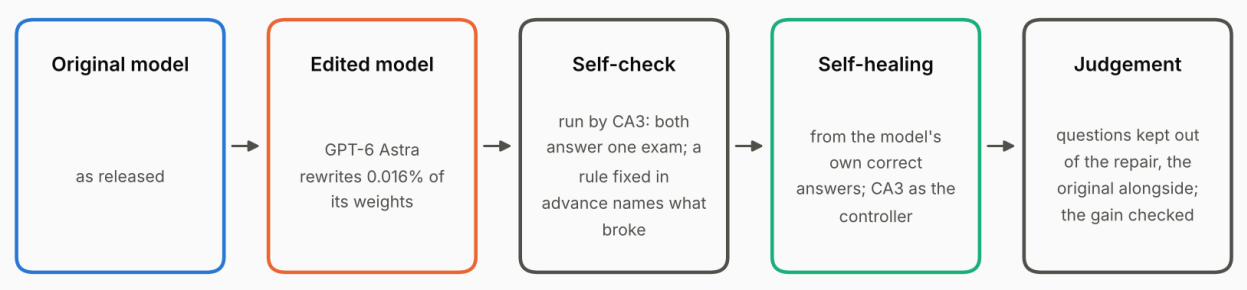


Figure 1: The procedure, in order. The original and the edited model answer one exam; a rule fixed in advance names what broke; the edited model is healed from its own correct answers, with CA3 as the controller; the judgement uses questions kept out of the healing, with the original alongside, and checks the gain.

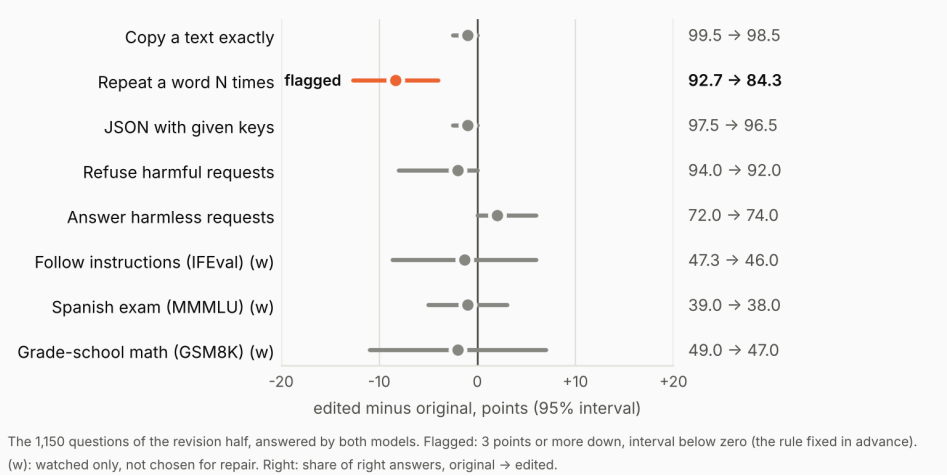


Figure 2: The self-check flags repetition and nothing else. Each of the eight skills, the edited model minus the original, with the 95% interval of the paired difference, on the revision half.

Can it heal itself without losing the gain?

EXPERIMENT 2

Can the edited model heal the skill from its own answers?

Next, we asked the edited model to heal repetition, with CA3 as its controller. The material is the model's own correct answers to revision questions of the broken skill: there is no outside teacher, and nothing is taken from the original. We judged the healing on 600 repetition questions that played no part in it, 300 of them drawn new before the run, with the original alongside.

SETUP

the edited model's own correct answers to revision questions of repetition

Heal repetition, and break nothing else.

No outside teacher, nothing from the original and no judgement question. CA3 keeps a write only if what the model already does right elsewhere stays within a fixed margin.

ANSWER

98.5% on 600 held-out repetition questions

up from 76.3% · before the edit, 92.0% · 98 of the 100 broken answers right again · 23 minutes on one NVIDIA A40

Repetition, 600 judgement questions	Right answers
Original, before the edit	92.0%
Edited by GPT-6 Astra [1]	76.3%
Self-healed with CA3	98.5%

The healing raised the edited model by 22.2 points (95% interval +18.7 to +25.8). Interestingly, it did not stop at the original: the self-healed model ends 6.5 points above the model the edit started from (interval +4.2 to +8.8). Of the 100 answers the edit had broken, 98 are right again (Figure 3).

How does it do it: CA3 writes the healing from the model's own correct answers, and keeps a write only if what the model already does right elsewhere stays within a fixed margin. How CA3 holds, places and controls what it writes is not described here.

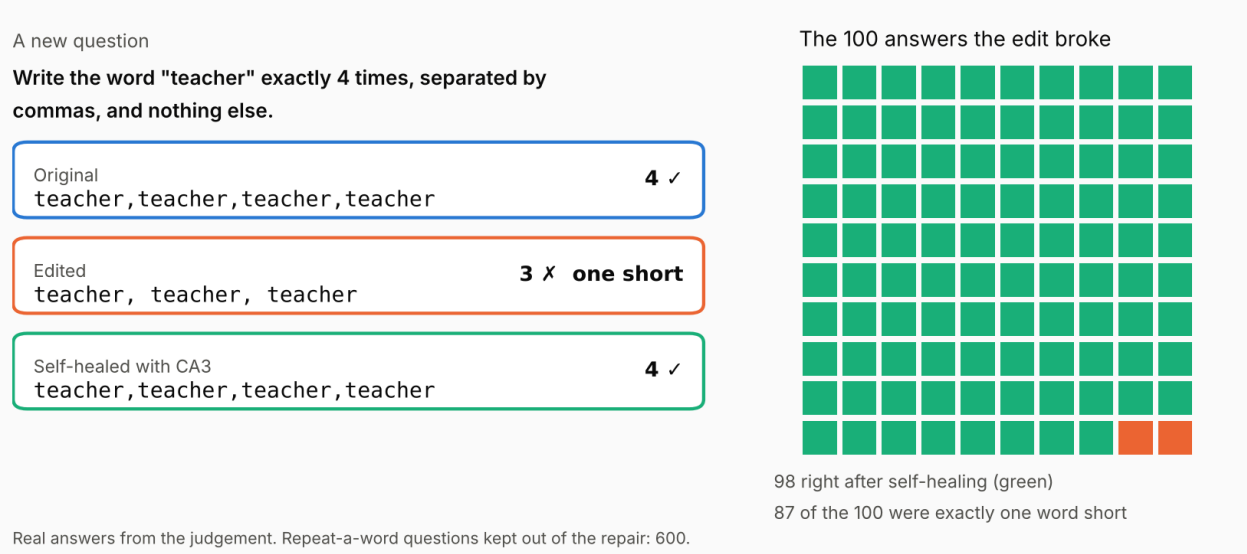


Figure 3: 98 of the 100 answers the edit broke are right after self-healing. Left, a new question with the three real answers. Right, the 100 answers the edit broke among the 600 judgement questions.

EXPERIMENT 3

Does the healing stay where it belongs?

A healing that spreads would trade one damage for another. So we looked at everything else: the judgement half of the seven other skills, and the math behind the edit's gain.

SETUP

the self-healed model, the judgement half of the seven other skills, and the 500 MATH-500 problems [15] and 1,119 GSM8K test problems [14] outside the exam

Does the healing reach anything but repetition?

The gain counts as kept if no more than 3% of either math set reaches the healing.

ANSWER

Nothing else moved

seven skills, 0.0 points of change · 0 of 500 MATH-500 and 0 of 1,119 GSM8K problems reach the healing · ✓ gain kept

CA3's control and routing kept the healing confined: none of the seven skills' questions reached it, and all 600 repetition questions did. The gain of the edit lies in math with thinking: MATH-500 from 7.9% to 48.1% and GSM8K from 13.8% to 67.1% in its authors' own evaluation [1]. None of the 1,619 math problems outside our exam reached the healing, so they are answered by the edited model as before, and the gain is intact (Figure 4).

Skill, judgement half	Edited	Self-healed with CA3	Change, points
Copying a text exactly	98.5%	98.5%	0.0
JSON with given keys	94.0%	94.0%	0.0
Refusing harmful requests	92.0%	92.0%	0.0
Answering harmless requests	76.0%	76.0%	0.0
Instruction following	54.7%	54.7%	0.0
Spanish exam	44.0%	44.0%	0.0
Grade-school math	53.0%	53.0%	0.0
Repetition, the healed skill	76.3%	98.5%	+22.2

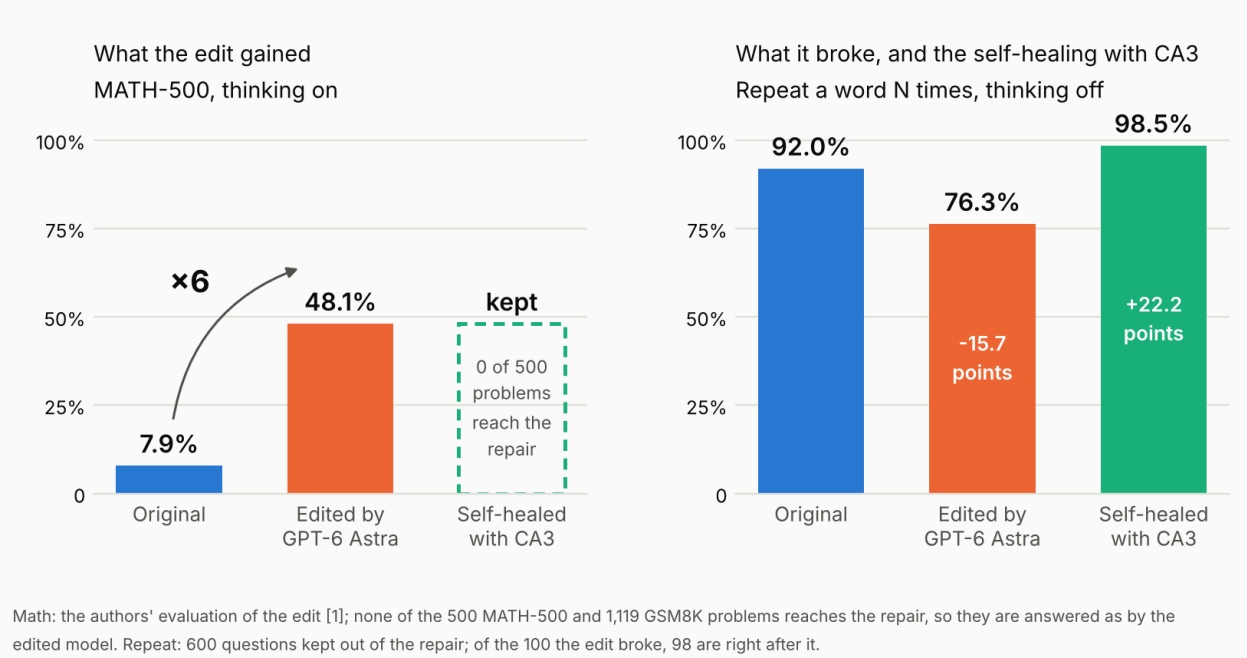


Figure 4: One edit, two effects, and the self-healing. Left, what the edit gained, as its authors measured it [1]: no math problem reaches the healing, so the gain is kept. Right, what the edit broke with thinking off, and the self-healing, on 600 questions kept out of it.

What we have not shown yet

This is one small model, one edit and one run, on a case where our earlier audit had shown what the edit broke: the model found it without being told, but we knew there was something to find. A self-check sees what its exam shows. With thinking off, on the questions kept for the judgement, the edit also scored below the original in instruction following; its authors, with thinking on, measured a threefold rise [1]. The self-check's half had not flagged it, and the healing left it as the edit left it. The healing was made and judged with thinking off, and the gain was checked by routing rather than by generating the long reasoning again.

Background and related work

Changes to a model can break what it did right. Model updates bring regressions on examples the old model got right [2, 3]; fine-tuning on harmless data can strip a model's refusals [4]; and knowledge edits degrade general abilities [5]. Learning from one's own correct answers is a known way to improve a model on a chosen task [16, 17, 18], but self-training can also raise its own metric and then collapse [8]. Without outside feedback, models also struggle to correct their own reasoning, and can make it worse [19]. Self-healing has been proposed for machine learning systems that work out why their performance degraded in deployment and correct it [20], and a theoretical framework for AI systems that detect and repair faults in their own reasoning [21]. Weight-Space Programming adds a new kind of change, written by a frontier model that reads the weights [1].

What is new here is a changed model that checks itself against its own past, finds what the change broke and heals it from its own answers, under a control that keeps the healing away from what the change improved. The task is not chosen by us: it is whatever the change broke.

What comes next?

In this post, we showed early signs that a changed model can heal itself: it finds what the change broke by comparing itself with its own past, and heals it from its own correct answers, while CA3 keeps everything else where it was. The results are early and limited, but they point to a different way of handling the changes every deployed model goes through. Here are some examples:

- A check-up after every change: a fine-tune on user feedback, a merge of adapters, a compression for a device, an edit proposed by an agent, a new release of a model a product depends on. Each is chosen for a gain; a self-check against the model's own past can tell what else it changed.
- Healing without a teacher: the material is the model's own correct answers, so no better model is needed to teach it.
- Small edits, checked: an edit that touches 0.016% of the weights can still break a skill, so even the smallest change is worth a check-up.
- Improve, check and heal: the authors of [1] see models that inspect, debug and improve their own weights as a richer path to recursive self-improvement. Self-improvement can also regress [8]; with a control, the loop can be an AI that proposes an edit, an edited model that checks itself against its past, and a healing of what broke while the gain stays.
- Homeostasis: biology's word for a body that changes and keeps its balance. Models that change all the time will need their own. That is the role we built CA3 for: the model heals itself, and CA3 makes sure nothing else breaks.

What comes next is that loop end to end. If you edit, merge or fine-tune models and want to know what your change broke, and get it healed without losing the gain, we would like to hear from you.

How we did it

Models. Qwen3.5-0.8B [10] at the revision the edit's authors used (2fc0636 on Hugging Face), in full precision, with thinking off. The edit (p50c, the one behind the sixfold gain) was rebuilt with the authors' own code and checked against their fingerprints [1]; it changes the MLP output weights of layers 15 and 19, as its authors describe. The original is the model as released.

Exam. Copying (400 questions), repeating (600), JSON (400), harmful and harmless requests (JailbreakBench, 100 and 100 [11]), instruction following (300 of IFEval's 541, official checkers [12]), Spanish multiple choice (200 of MMMLU [13]) and grade-school math (200 of GSM8K [14]). Each part was split once into a revision half, the only questions the model used to check itself, and a judgement half, kept apart until the end; for the generated measures, 300 more revision and 300 new judgement questions were drawn before anything was answered.

Rule. The comparison, the choice of the skill, the size of the healing and every verdict were fixed in a pre-registration before any measurement. Differences are paired by question, with 2,000 bootstrap resamples (seed 0) and the 95% interval.

Healing. CA3 wrote the repair from the edited model's own correct answers to revision questions of the chosen skill, under its control, which keeps a write only if the model's answers to what it already does right elsewhere stay within a fixed margin. How CA3 holds, places and controls what it writes is not described here.

The gain. The 500 problems of MATH-500 [15] and the 1,119 GSM8K test problems [14] outside the exam were put through the self-healed model's routing; the gain counts as kept if no more than 3% of either reach the healing.

Checks. Two rounds of independent code review before the run, with no blocking findings; a local test of 80 checks passed on our machine and again on the pod. The run took 23 minutes (1,350 s) on one NVIDIA A40, with a peak of 24.9 GB of GPU memory.

Credit

The edits and their evaluation are the work of Takehi, Teoh, Zhang and Khattab [1], released with their logs and code under CC BY 4.0. That release is what made an independent check possible.

Intellectual property

CA3 is the subject of patent applications by Biome Makers Inc.

Use of AI assistance

The experimental code, the analyses and the drafting of this text were done with Claude (Anthropic) under the author's direction. The author designed the study, fixed the pre-registration, reviewed every result and takes responsibility for the content.

Citation

Please cite this post as

Alberto Acedo. "Sparks of Self-Healing: Early experiments with CA3." Biome Makers Inc., November 2026. [link once published]

or in BibTeX

```
@misc{acedo2026selfhealing,
  title        = {{Sparks of Self-Healing: Early experiments with CA3}},
  author       = {Acedo, Alberto},
  year         = {2026},
  month        = nov,
  howpublished = {[link once published]},
  note         = {Blog post, Biome Makers Inc.}
}
```

References

[1] Takehi R., Teoh J., Zhang A. and Khattab O. Sparks of Weight-Space Programming: LLMs can natively read & write neural network weights. MIT CSAIL, blog post, 2 November 2026. https://rikiyat.github.io/blog/weight-space-programming/ (edits and logs, CC BY 4.0: https://github.com/RikiyaT/lm_understand-logs)

[2] Xie Y. et al. Regression Bugs Are In Your Model! Measuring, Reducing and Analyzing Regressions In NLP Model Updates. ACL 2021. https://aclanthology.org/2021.acl-long.515

[3] Echterhoff J., Faghri F., Vemulapalli R., Hu T.-Y., Li C.-L., Tuzel O. and Pouransari H. MUSCLE: A Model Update Strategy for Compatible LLM Evolution. 2024. https://arxiv.org/abs/2407.09435

[4] Qi X. et al. Fine-tuning Aligned Language Models Compromises Safety, Even When Users Do Not Intend To! ICLR 2024. https://arxiv.org/abs/2310.03693

[5] Gu J.-C. et al. Model Editing Harms General Abilities of Large Language Models: Regularization to the Rescue. EMNLP 2024. https://arxiv.org/abs/2401.04700

[6] Claburn T. Forget retraining - AI can now tweak another AI's weights. The Register, 3 November 2026. https://www.theregister.com/a/5307105

[7] swyx. LLMs can read & write neural network weights: "Weight Space Programming" cuts Autoresearch out of the RSI loop. Latent Space (AI News), 3 November 2026. https://latent.space/p/ainews-llms-can-read-write-neural-network-weights-weight-space-programming-cuts-autoresearch-out-of/

[8] Lin J. Self-Improvement Can Self-Regress: The Rise-and-Collapse Failure Mode of LLM Self-Training. 2026. https://arxiv.org/abs/2606.21090

[9] Acedo A. CA3 audit of the published weight edits of [1] (gates 1 and 2). Biome Makers Inc., November 2026, internal report.

[10] Qwen Team. Qwen3.5 small models (0.8B, 2B, 4B, 9B). 2026. https://huggingface.co/Qwen/Qwen3.5-0.8B

[11] Chao P. et al. JailbreakBench: An Open Robustness Benchmark for Jailbreaking Large Language Models. NeurIPS 2024 Datasets and Benchmarks. https://arxiv.org/abs/2404.01318

[12] Zhou J. et al. Instruction-Following Evaluation for Large Language Models. 2023. https://arxiv.org/abs/2311.07911

[13] OpenAI. MMMLU: Multilingual Massive Multitask Language Understanding. 2024. https://huggingface.co/datasets/openai/MMMLU

[14] Cobbe K. et al. Training Verifiers to Solve Math Word Problems. 2021. https://arxiv.org/abs/2110.14168

[15] Lightman H. et al. Let's Verify Step by Step. 2023. https://arxiv.org/abs/2305.20050 (the MATH-500 subset of Hendrycks D. et al., Measuring Mathematical Problem Solving With the MATH Dataset, NeurIPS 2021, https://arxiv.org/abs/2103.03874)

[16] Zelikman E., Wu Y., Mu J. and Goodman N. STaR: Bootstrapping Reasoning With Reasoning. NeurIPS 2022. https://arxiv.org/abs/2203.14465

[17] Singh A. et al. Beyond Human Data: Scaling Self-Training for Problem-Solving with Language Models. 2023. https://arxiv.org/abs/2312.06585

[18] Zweiger A. et al. Self-Adapting Language Models. 2025. https://arxiv.org/abs/2506.10943

[19] Huang J., Chen X., Mishra S., Zheng H.S., Yu A.W., Song X. and Zhou D. Large Language Models Cannot Self-Correct Reasoning Yet. ICLR 2024. https://arxiv.org/abs/2310.01798

[20] Rauba P., Seedat N., Kacprzyk K. and van der Schaar M. Self-Healing Machine Learning: A Framework for Autonomous Adaptation in Real-World Environments. NeurIPS 2024. https://arxiv.org/abs/2411.00186

[21] Figurelli R. Self-HealAI: Architecting Autonomous Cognitive Self-Repair. Preprints.org, June 2025. https://www.preprints.org/manuscript/202506.0063