Preprint · November 2026

The Invoice

Data, Energy and Agency Taxes of Transformer Intelligence

Paulo Câmara Institute of Neuro-Humanities & Science (INHS)

Data, code and power traces: doi:10.5281/zenodo.24204647

Abstract. Transformers work. I ask what they charge for working, and which part of the bill comes from the design. I call such a charge a tax and price four. Building the learner: by published estimates, three models trained between 2020 and 2025 consumed the food energy of 76 to about 18,000 human upbringings, and three and a half to five orders of magnitude more language than a child hears. Learning after deployment: a LoRA run that taught a 7B model 200 invented facts cost 11.9 kJ, after which it recalled 33.5% forward and none in reverse; kept in context without caching, the facts came back 78% both ways at up to 21 times the energy per query. A fixed price per token: familiar and unfamiliar instances cost the same energy per token (ratios 0.98 to 1.05 from 3B to 32B), so the bill is set by length, and deriving a new count took frontier models 64 to 121 times the tokens of reciting a known one. Agency: a brain arrives with it, and a model buys it in re-read context, at a measured 830 Wh for one software task in the literature. I set these against a ledger with the same boundary on both sides. Per word of conversation a brain and a served chat model cost the same order of magnitude; the brain wins on the marginal cost of thinking, on copies and on formation, and loses to the model on serial counting. I then score the alternatives, from neuromorphic chips to one-bit weights and continual-learning memories, on what a meter has shown, give an energy budget for a design that pays for activity, and propose five requirements with their tests, led by joules per verified correct answer.

Keywords: energy per verified answer · inference and training energy · continual learning · reversal curse · agents · sparse coding · neuromorphic computing

1 Nobody brings the invoice

For years the argument about large language models has been a tennis match over whether they reason, and both players keep score in benchmarks. In all that time I have not seen either of them ask what the court costs.

I bring the invoice, in joules, because the joule has no opinion about philosophy. A brain that solves a problem and a GPU that solves the same problem both turn energy into an answer, and the conversion can be estimated on both sides.

A tax, in this paper, is a charge the design makes whether or not the task needs it. A hard problem is allowed to be expensive. A tax is what you pay on the easy ones too. I price four of them. Two I measured on my own metered GPU, one I assembled from disclosed figures, and the fourth, agency, I take from measurements that others published in 2026.

An earlier draft of this paper said that a trained model serves millions of people and a brain serves one. That sentence was wrong, and correcting it reorganised the paper. A teacher serves hundreds of pupils in a career, a recorded lecture serves millions, and a book serves whoever opens it for as long as paper lasts. What a brain produces can be fixed once and copied for the price of the medium. What a model produces is generated again for each person who asks. And a brain comes with a body, senses and a memory it does not have to re-read, which is most of what the industry now sells separately under the name of agents. So the comparison needs a ledger with the same boundary on both sides, and Section 3 builds one. The brain does not win every line of it.

I wrote this for people who build these systems. Section 2 shows the bill and how fast it is growing. Section 3 draws the same boundary around a brain and a model. Sections 4 to 7 price the four taxes, with intervals and raw data. Section 8 keeps the results that go against my own thesis, because a paper that reports only

the taxes it found is an advertisement. Section 9 checks who claims to pay less against a meter, and Section 10 turns what is left into a budget and five tests.

Inference energy of large models has been measured before (Samsi et al., 2023; Luccioni, Jernite & Strubell, 2024; Elsworth et al., 2025; Chung et al., 2026). What I add is bookkeeping. Everything is in joules, no comparison crosses a boundary, and each way the brain has of dodging a charge is rewritten as a test a machine can fail.

2 The bill at the scale of a planet

One answer is cheap. Google measured the median text prompt of its assistant at 0.24 Wh, counting the accelerator, the host machine, idle capacity and the building (Elsworth et al., 2025). A two-kilowatt kettle spends that in under half a second. The number that matters is what happens when a cheap thing is sold by the quadrillion.

In May 2024 Google processed 9.7 trillion tokens a month across its products. A year later the figure was about 480 trillion, and in May 2026 it passed 3.2 quadrillion (Pichai, 2026). That is a factor of 330 in two years, and it works out to roughly 13,000 tokens per day for every person alive (Figure 1). On a public routing service, weekly tokens grew 250-fold in twenty months (OpenRouter, 2026). I found no official message or token volume from the other large providers after July 2025, when ChatGPT was serving 18 billion messages a week to 700 million users (Chatterji et al., 2025).

Efficiency improved over the same period, and volume outran it. Google reports that the energy of its median prompt fell 33-fold between May 2024 and May 2025. In those twelve months its token volume rose 49.5-fold. The two series have different bases, one a median prompt of one product and the other tokens on every surface, so the pair is compatible with a rebound effect and does not prove one (Luccioni, Strubell & Crawford, 2025). The company's total electricity demand grew 37% in 2025 (Google, 2026). Prices tell the same story from the buyer's side: the price of a fixed level of capability falls five- to tenfold a year, while the cost of running the best available model rises three- to eighteenfold a year, because the frontier moves to larger models that think longer (Gundlach et al., 2025). Table 1 puts the bill in ten lines.

Table 1. The bill in ten lines. Each number is the source's own, with its boundary; ratios marked "my arithmetic" are mine.

Quantity	Value	Comparison	Source
Tokens processed by Google per month	3.2×10^{15} (May 2026)	9.7×10^{12} in May 2024, 330× in two years (my arithmetic)	Pichai (2026)
Energy of a median text prompt	0.24 Wh, full serving stack	33× lower than a year earlier	Elsworth et al. (2025)
Electricity of data centres, world	415 TWh in 2024; about 485 TWh in 2025, of which 155 TWh AI-focused	945 TWh projected for 2030, about 3% of world electricity	IEA (2025) for 2024 and 2030; Ritchie (2026), citing the IEA, for 2025
Largest training run with disclosed power and duration	237 GWh (Grok 3, 2025; ceiling from power × time)	1.8 GWh for GPT-3 in 2020, same method	Epoch AI (2026a), my arithmetic
Power of the largest training runs	growing 2.2× a year, above 100 MW in 2025	4 to 16 GW projected for 2030	You & Owen (2025)
Largest AI data centre, IT power	1,037 MW (November 2026)	about 300 MW in March 2025	Epoch AI (2026b); Pilz et al. (2025)
Reasoning against plain generation	30× the GPU energy on average, 150× to 700× within one model	chat = 1	Luccioni & Gamazaychikov (2025)
Share of Ireland's metered electricity	23% in 2025	5% in 2015; households 28%	Central Statistics Office (2026)
PJM capacity auction, 2028/29	US$ 6.27 bn of 16.41 bn attributed to data-centre load	US$ 29.4 bn over four auctions	Monitoring Analytics (2026)
Microsoft, electricity and emissions	electricity × 3.44, total emissions +62% (market-based), FY2020 to FY2025	location-based scope 2 × 2.78	Microsoft (2026)

Training is the part of the bill that is paid before the first answer. Among the models in Epoch AI's database that disclose both power and duration, the largest training run of each year went from 1.8 GWh in 2020 to

237 GWh in 2025, and a fit to the yearly maximum since 2012 grows about 3.1-fold a year (Figure 2). The series is ragged. A game-playing system of 2019 sits above the 2020 point, and the 2024 and 2026 maxima fall because the largest models of those years published no training time. These are ceilings, power at full draw for the whole run, computed with Epoch's power estimates, which is why GPT-3 appears here at 1.8 GWh and in Section 4 at another author's 1.3 GWh. After 2023 they are also a floor on the frontier. Epoch's own analysis has the power of the largest runs growing 2.2-fold a year, past 100 MW in 2025, and reaching 4 to 16 GW by 2030, while the energy efficiency of the hardware improves about 40% a year (You & Owen, 2025).

The world total looks modest. Data centres used about 415 TWh in 2024, around 1.5% of the world's electricity, and the International Energy Agency writes that concerns about AI accelerating climate change "appear overstated" (IEA, 2025). Koomey et al. (2026) warn that load forecasts tend to overestimate, in part because the same project requests a grid connection in several places. I take both points. The bill is local before it is global. Data centres took 23% of Ireland's metered electricity in 2025, against 28% for all its homes (Central Statistics Office, 2026). The independent monitor of the largest American grid attributes 38% of the capacity-auction revenue for 2028/29 to data-centre load, existing and forecast, a counterfactual estimate that it does not present as an invoice (Monitoring Analytics, 2026). Water follows electricity: in the United States a kilowatt-hour delivered to a data centre consumes about 4.9 litres, 0.4 on site and 4.5 at the power plant (Shehabi et al., 2024; the sum is my arithmetic).

A comparison with brains helps fix the scale, and I use it with a warning. Eight point two billion people (United Nations, 2024) at 20 W each think on about 1,440 TWh a year. The data centres of 2025 drew a third of that, and the AI-focused ones alone drew the power of roughly 885 million brains. As energy policy this image is weak, since nobody proposes replacing servers with people. It turns into an argument only when the cost of each correct answer enters, and that is what the rest of the paper tries to supply.

3 A ledger with the same boundary on both sides

Comparisons between brains and machines usually cheat by moving the boundary. One famous result charges the human writer with their whole annual carbon footprint, holidays and heating included, and charges the model with a training run divided by an assumed 300 million queries; the model then wins by a factor of 130 to 1,500 per page (Tomlinson et al., 2024). The choice of system boundary was criticised at the time as implausible (Pauliuk, 2024). The opposite cheat is the one I committed in earlier drafts: a 20 W brain against a training run, as if the brain had not needed twenty years of meals to become one. I use three boundaries and compare only within each.

What it costs to form a person. A child's brain is at its most expensive around age five, when it takes up 167 g of glucose a day in boys, 1.9 times the adult figure; as a share of resting metabolism it peaks a year earlier, at 66% (Kuzawa et al., 2014). At birth the whole organ runs on about a watt (my conversion of the oxygen consumption measured by Liu et al., 2014). These are glucose taken up, an upper bound on energy, and the adult figure they imply is about 16 W. Summing the energy requirements published by the FAO for each year of life from birth to twenty gives about 17 MWh of food energy for the whole body, 15.5 for girls and 18.4 for boys (FAO/WHO/UNU, 2004; the sum is mine). Between 2.8 and 5.3 MWh of that passes

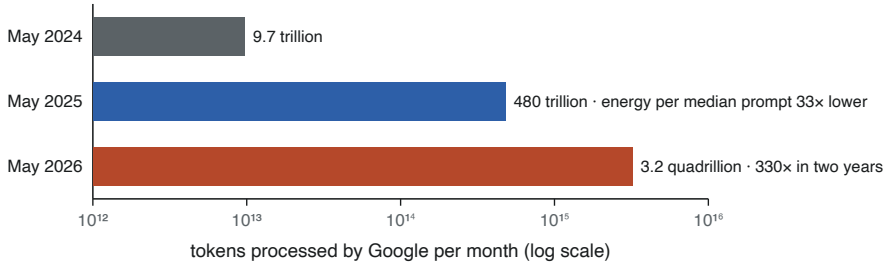

Figure 1. Tokens processed per month across Google's products, as stated by the company (Pichai, 2026). Over the first interval the energy of a median text prompt fell 33-fold (Elsworth et al., 2025) and the token volume rose 49.5-fold. The two series have different bases. Log scale.

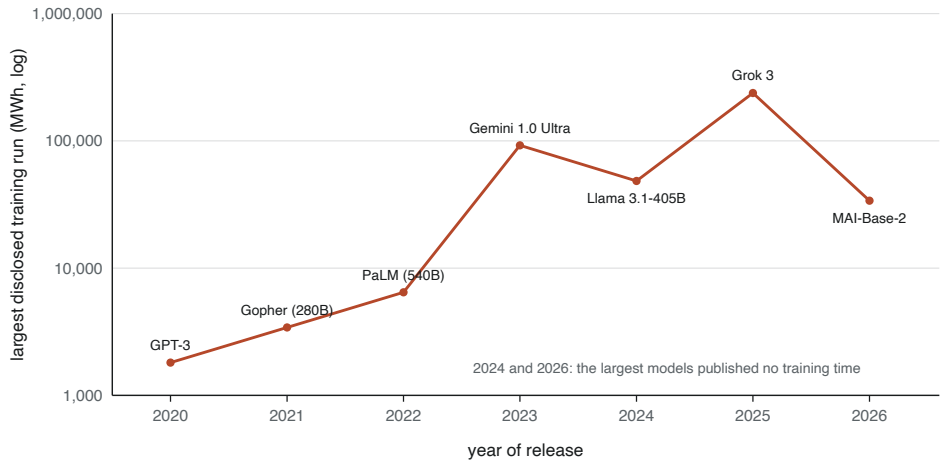

Figure 2. Largest training run of each year among the models in Epoch AI's database that disclose both power and duration (153 of 1,092): estimated power × stated time, a ceiling. A fit to the yearly maximum from 2012 to 2026 grows about 3.1-fold a year. The dips are missing disclosures. Log scale.

through the brain, depending on whether one takes Kuzawa's adult rate or the childhood peak for the whole period; the round 20 W used elsewhere in this paper gives 3.5. I will call 17 MWh one *formation*. It is food energy set against electricity, it stops at twenty, and it counts metabolism only. It leaves out growing the food, the house, the school and the teachers' own formations, the same way my machine figures leave out chip fabrication, failed experiments and the people who wrote the training corpus. A thirty-year career adds about 26 MWh of body to the human side, and I leave that out too, since it belongs to operation.

What a brain serves. The classroom teacher is the expensive case: at the world average of 23.4 pupils per primary teacher (UNESCO, via World Bank, 2018) and an assumed thirty-year career, about 700 children. A primary-care physician's conventional panel is about 2,500 patients, a figure Altschuler et al. (2012) take as their baseline and consider too large. One recorded course by one lecturer, first offered in 2012, has been taken by more than 4.8 million people (DeepLearning.AI, 2026). One author's series has sold over 500 million copies (Wizarding World, 2018). The economics literature has the right word for this. An idea is a non-rival good: one person's use does not consume it, and the cost is paid once (Romer, 1990). The weights of a trained model are non-rival in exactly this sense (Korinek & Vipra, 2025). An answer is not. Each answer is produced to order, at a cost in joules, for one reader. A person does both: answers to order in a classroom or a clinic, and fixes what they know in an artefact that copies for the cost of paper or a download, which I did not measure, plus the reader's own 5 J per word read. I have not found this asymmetry stated in the academic literature on the economics of AI, so I state it as mine: the model is sold as the book and billed as the tutor.

The three boundaries. At the margin, thinking is nearly free for a brain. The extra energy associated with a change in brain activity is "often less than 5%" of the baseline (Raichle, 2015), under one watt, and the fact that mental arithmetic leaves whole-brain metabolism almost unchanged has been known for seventy years (Sokoloff et al., 1955, as cited by Raichle). A narrator speaks about 160 words a minute (Zheng & Meister, 2025), so the marginal cost of a spoken word is about 0.4 J. In operation, with the whole organ on the bill, the same word costs 7.5 J, and with the whole resting body at 100 W, 37.5 J. A chat model in operation is in the same range. Google's 0.24 Wh is 864 J per prompt; if a reply runs to 300 words, an assumption of mine, that is 2.9 J a word, and with 100 words it would be 8.6 J. On the operating line, per word of plain conversation, the brain alone and the fully loaded serving stack are within an order of magnitude of each other, and against the whole body the model is the cheaper one (Table 2). I will call that a tie and not lean on it.

Table 2. One ledger, three boundaries. Human figures use 20 W for the brain, 100 W for the body and 160 spoken words a minute. The per-word figure for the chat model assumes 300 words per reply. Formation and training are divided by people served only in Table 3.

Boundary	Brain	Transformer	Cheaper side
Marginal: one more unit of thought	under 1 W above baseline; about 0.4 J per spoken word; zero for one more reader of a book	every answer generated again: 864 J per median chat prompt; 4× to 15× the tokens for agents	brain
Operating: everything that runs during the work	7.5 J per spoken word (brain), 37.5 J (resting body); a full active day of a body, 2.9 to 4.0 kWh	about 2.9 J per word of chat; a heavy day of a coding agent, about 3.0 kWh (estimate)	tie in chat; roughly even for a day of agent work
Loaded: formation or training	17 MWh of food energy per adult (about 3.5 MWh through the brain)	1,287 to about 310,000 MWh per training run, repeated each generation	brain, by 76× to 18,000×

The loaded line changes when it is divided by the people served, and here the model's reach counts in its favour (Table 3, Figure 3). A teacher's formation spread over 700 pupils is 24 kWh per pupil. A training run spread over 700 million weekly users, a reach I borrow from one product and apply to models whose energy was published by others, is 1.8 Wh per user for GPT-3 and about 440 Wh for Grok 4. The services are not the same, since a teacher gives each pupil a year and a model gives each user some answers, and none of these three models served that many people. Taken as orders of magnitude, the model beats the classroom teacher and the physician. It loses to the recorded lecture and to the book, which reach comparable numbers with one formation and pay nothing per reader. And the model's line has a clause the human line lacks: the next generation is trained afresh, and nothing is inherited.

Table 3. Formation or training energy per person served. Human reach: sources in the text; a thirty-year teaching career is my assumption. Model reach: 700 million weekly users, a figure from one product (Chatterji et al., 2025) applied to three models as an illustration of order of magnitude.

Who serves	Energy to form or train	People served	Energy per person served
Primary teacher	17 MWh	about 700 pupils in a career	24 kWh
Primary-care physician	17 MWh	conventional panel of 2,500	6.8 kWh
Lecturer, recorded course	17 MWh	4.8 million learners	3.5 Wh
Author, book series	17 MWh	500 million copies	0.03 Wh
GPT-3 (2020)	1,287 MWh	700 million	1.8 Wh
Llama 3.1 405B (2024)	about 21,600 MWh	700 million	31 Wh
Grok 4 (2025)	about 310,000 MWh	700 million	440 Wh

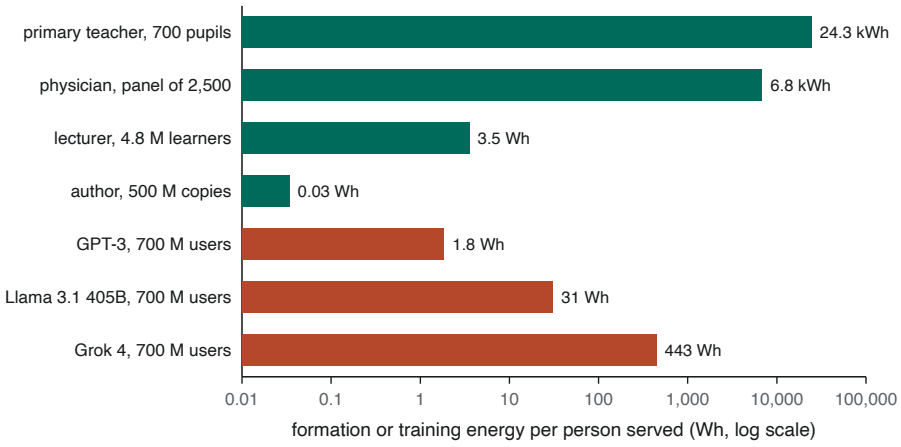

Figure 3. Formation or training energy divided by people served (Table 3). Green: one human formation of 17 MWh spread over the reach of four occupations. Rust: three training runs spread over 700 million weekly users, a reach borrowed from one product. Log scale.

One more entry belongs on the model's side and is never booked. The training corpus is the fixed output of brains, written by people who each cost a formation. The model reads for free what was expensive to produce, which is the non-rival good working as Romer described. It also means the first tax below understates what the learner consumed.

4 The first tax: building the learner

Nobody trains a child. A child is exposed. By age thirteen the exposure comes to fewer than 100 million words, a rounded-up budget, at 2 to 7 million a year, counting all the speech around them, including what they only overhear (Warstadt et al., 2023, citing Gilkerson et al., 2017). The figure varies tenfold or more across cultures and households, and I use the upper end, the one least favourable to my argument. Frank (2023) puts the gap between children and language models at four to five orders of magnitude.

The standard recipe discloses its corpus: 300 billion tokens for GPT-3 (Brown et al., 2020), 15.6 trillion for Llama 3 405B (Llama Team, 2024), 18 trillion for Qwen2.5 (Qwen Team, 2024), 14.8 trillion for DeepSeek-V3 (DeepSeek-AI, 2024). In Epoch AI's database the largest disclosed language corpus of each year went from about 10^{12} tokens in 2020 to 4.5×10^{13} in 2026. A token is not a word. How many tokens make a word depends on the tokenizer and the language, so the ratios below are orders of magnitude and nothing finer.

Energy is harder, because few developers publish it. Where they do, I use their figure. Where they publish GPU-hours and device power, I multiply the two and call the result a ceiling (Table 4; Figures 4 and 5). Estimates differ with the assumptions: Epoch AI's database puts the same Llama run at about 48 GWh with its own power figure, and I keep the narrower ceiling.

Table 4. Energy and language consumed to build three disclosed systems. A formation is 17 MWh, the food energy of a human body from birth to twenty; a twenty-year brain is 20 W for 20 years, 3.5 MWh. The child hears fewer than 10^8 words by age 13.

System	Training energy	Human formations	Twenty-year brains	Training tokens vs. a child's words
GPT-3 (2020)	1,287 MWh, estimated (Patterson et al., 2021)	76	367	3,000×
Llama 3.1 405B (2024)	21,600 MWh, 30.84 M H100-hours × 700 W TDP, no facility overhead (Llama 3.1 model card)	1,270	6,200	156,000×
Grok 4 (2025)	310,000 MWh, third-party estimate (Epoch AI, 2025)	18,200	88,000	not disclosed

The 20 W is the usual figure for the glucose budget of the whole organ, from classical measurements of glucose uptake (Balasubramanian, 2021), and it is an upper bound: a recent PET-based estimate puts it at 17 W (Levy & Calvert, 2021). Levy and Calvert estimate that the cortex's computation proper costs about 0.1 W, with communication costing 35 times more. With that figure the brain column would grow about two hundredfold, but I would be dropping the brain's signalling while still charging the GPU for its memory traffic. I leave it out.

The honest objection to the word count is that a child receives far more than words. LeCun's version is that the optic nerves of a four-year-old have carried about 10^{15} bytes in 16,000 waking hours, fifty times the text of a large training corpus (LeCun, 2024). The brake on that argument comes from the same field: the senses take in about 10^9 bits a second and behaviour comes out at about 10 (Zheng & Meister, 2025), so raw bytes are a poor measure of what was learned. What the child certainly has is structure that arrives before the data. Infants represent objects, actions, number and space with systems that do not wait for instruction (Spelke & Kinzler, 2007), and most animal behaviour is compressed into wiring rules by the genome (Zador, 2019). A newborn's whole brain runs on about a watt, and the fetus was already sucking its thumb. That is behaviour without a training set, and evolution's search for it has already been paid for and published.

The BabyLM challenge trains language models inside the child's budget, and its 2025 round is the most useful data point I know on this tax: with 100 million words some systems match a 70-billion-parameter model on a grammar benchmark and still fall short of humans on language understanding (Charpentier et al., 2025). In the 2024 round the score tracked training compute, which means the entries bought data efficiency

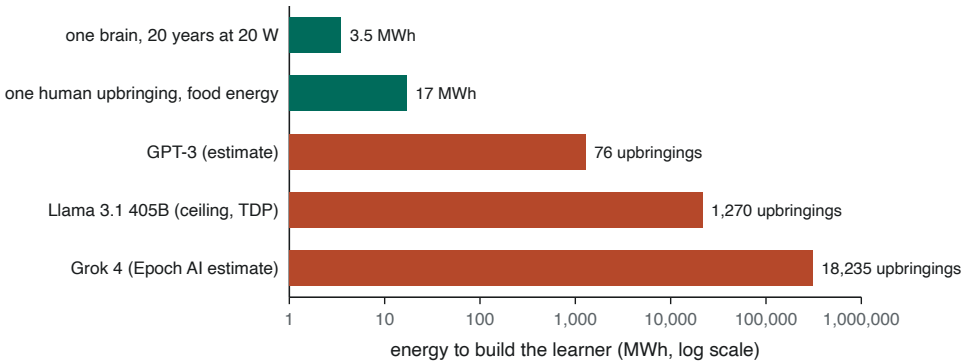

Figure 4. Energy to build the learner. A human upbringing is 17 MWh of food energy from birth to twenty. GPT-3 is a third-party estimate (Patterson et al., 2021); Llama 3.1 405B is a ceiling from disclosed H100-hours × 700 W TDP without facility overhead; Grok 4 is a third-party estimate (Epoch AI, 2025). Log scale.

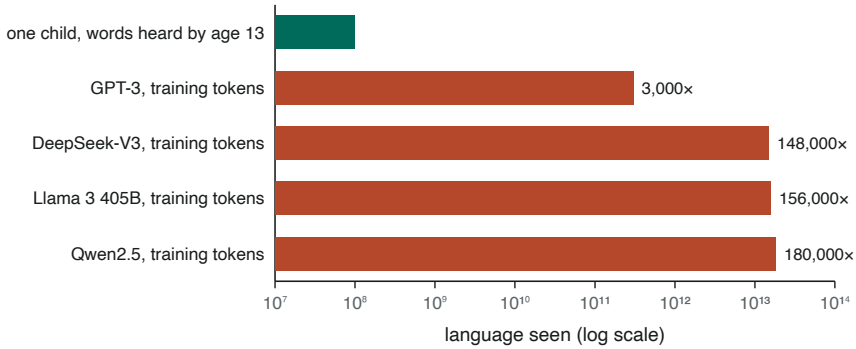

Figure 5. Language seen before the first answer. Child: fewer than 10^8 words by age 13, the upper end of a range that varies tenfold (Warstadt et al., 2023). Models: disclosed training tokens. A token is not a word; ratios are orders of magnitude.

with more passes (Hu et al., 2024). A child's diet of words is enough to buy grammar. Understanding is still being bought with compute, and the receipt is in the 2024 round.

Two cautions sit next to Table 4. The training energy is paid once and then spread over every query the model ever serves, and Table 3 shows how small that can be per user. It is the price of acquiring competence, and it measures how wastefully the recipe learns. And the next generation pays it again.

5 The second tax: learning after deployment

You learn a colleague's name over lunch and still know your own address after dessert. The standard account of how that works is two memory systems: a fast hippocampal store that takes in new items without disturbing old ones, and a slow neocortical store that absorbs structure through interleaved replay (McClelland, McNaughton & O'Reilly, 1995; updated by Kumaran, Hassabis & McClelland, 2016), much of it during sleep (Klinzing, Niethard & Born, 2019). Fast acquisition of arbitrary associations is real. In Sharon, Moscovitch & Gilboa (2011) it survived hippocampal damage in four patients, which suggests more than one route in. The adult effect has failed to replicate in some paradigms (Greve, Cooper & Henson, 2014; Gurunandan et al., 2023), so what I claim is fast acquisition, not one-shot learning. People also tend to store a pair as a single association. In Kahana's analysis of paired-associate recall, forward and backward recall of the same pair correlate almost perfectly (Kahana, 2002).

A deployed transformer has frozen weights and two options. It can keep the new fact in its context window and pay for it again on every query. Or it can be fine-tuned, which is known to erode what it knew before (McCloskey & Cohen, 1989; Kirkpatrick et al., 2017; Luo et al., 2023 for LLMs between 1B and 7B parameters). A model fine-tuned on "A is B" also fails to conclude that "B is A", the reversal curse (Berglund

et al., 2024). In the sources I checked I found no measurement of what it costs, in energy, for a language model to learn a fact this way, or of what the learning pushes out.

So I measured it. The model was Qwen2.5-7B-Instruct in bf16 on one L40S. LoRA r = 16, $\alpha = 32$, dropout 0.05, all linear layers, learning rate 2×10^{-4}, three epochs, batch 8, all fixed in a prediction file registered before the run. The training loss covered every token, the user turn of the question–answer pairs included. The facts were 200 invented lakes, each in its own invented province, in Portuguese. Training data: 1,000 sentences (five phrasings per fact, with the lake always before the province) and 200 forward question–answer pairs worded differently from the test. Retention: 300 MMLU questions in English, scored by the most probable answer letter. Energy is GPU energy above the idle baseline of 64 W. Answers were cut at 20 tokens. Queries without context ran in batches of 16, and queries carrying the facts in context in batches of 4.

Table 5. Teaching Qwen2.5-7B-Instruct 200 invented facts: recall in both directions, retention on MMLU and GPU energy, before and after one LoRA run.

Phase	Forward (lake → province)	Reverse (province → lake)	MMLU (300)	GPU energy, net
Before, no context	0%	0%	75.0%	4.5 J per query
Before, all 200 facts in context	78.5%	78.0%	—	96 J per query
LoRA training	—	—	—	11.9 kJ (79 s, 85,833 tokens)
After, no context	33.5%	**0 of 200**	72.3%	2.7 J per query

The query after training was cheaper than the query before it, 2.7 J against 4.5 J, and I did not investigate why.

Table 5 and Figures 6 and 7 describe two ways of paying for the same facts.

The context window works. The model answers in both directions about 78% of the time. The price is a preamble with all 200 facts in front of every question, which took each query from 4.5 J to 96 J, 21 times more. Read that ratio as an upper bound, since the context queries ran in batches four times smaller, and caching the preamble or retrieving only the relevant fact would bring it down. Providers price a cached read at about a tenth of fresh input, and the cache expires within minutes to an hour (Anthropic, 2026). I did none of that. Either way, nothing was learned. The fact is re-read, and re-billed, every time it is used.

Training the weights cost 11.9 kJ once, what a 20 W brain spends in ten minutes. As measured, the context route's extra cost catches up with that after about 130 queries, a number that would grow with equal batching or with caching. On energy, the weight route is the cheaper one. The tax shows up in what the money bought. The model kept a third of the facts in the direction it was trained and none in the other. Reverse recall was 0 of 200, which replicates the reversal curse and adds its price. MMLU fell by 8 of 300 questions, net. The same questions were asked before and after, so the right test is a paired one, and I did not store per-

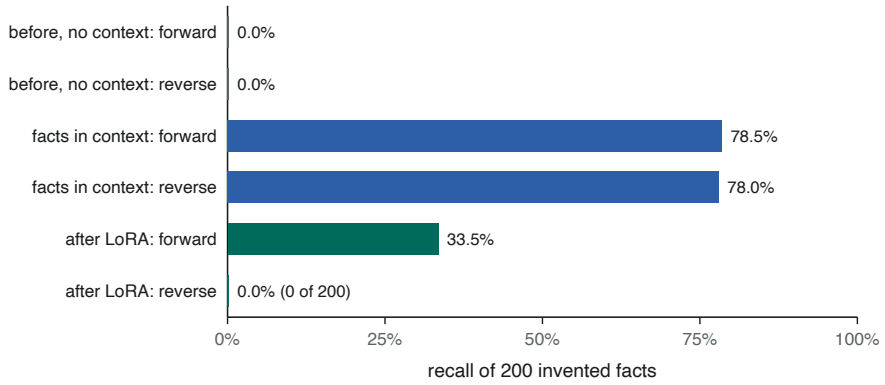

Figure 6. Recall of 200 invented facts by Qwen2.5-7B-Instruct, forward (lake → province) and reverse (province → lake), before training, with all facts in the context window, and after a LoRA run trained in the forward direction only. Each bar is 200 questions.

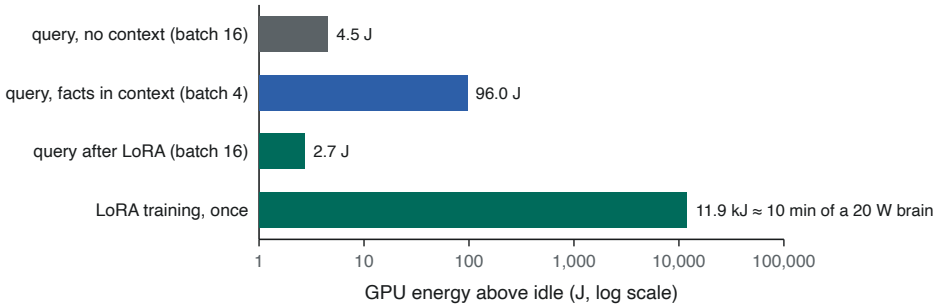

Figure 7. GPU energy above the 64 W idle baseline, NVML integral at 10 Hz. Per query: mean over 400 questions, answers cut at 20 tokens; the context queries ran in smaller batches, which inflates their cost. The context route pays on every query; the weight route pays once and learns less.

question results. If all eight changes were losses, McNemar's test would give $p \approx 0.008$. With eleven losses and three gains it would give $p \approx 0.06$. The data do not decide between those.

My invented names were too alike. Thirty-one pairs of provinces and 98 pairs of lakes differ by two letters or fewer. After training, 71 of the 200 forward answers named a trained province, but the wrong one. That is more than the 67 that were right. The model learned the vocabulary of provinces and got the pairs wrong more often than right, and the similarity of the names inflates both kinds of error by an amount I cannot separate.

Of the seven quantities I registered before this run, three held and four missed. Three of the misses overestimated what the model would learn, and the fourth overestimated what learning would cost. I expected forward recall of at least 80% after training (it was 33.5%), in-context recall of at least 90% forward (78.5%) and 80% reverse (78.0%), and training energy between 50 and 500 kJ (11.9 kJ). Recall before training (zero both ways), reverse recall after training (at most 10%) and an MMLU change of one to five points all held.

This is one run, one model, one configuration and one seed. A larger model, more epochs or training data written in both directions would raise these numbers. Writing every fact in both directions is the obvious workaround for the reversal curse. It amounts to writing each fact twice, because this learner, trained this way, inverted none of them on its own.

The literature since 2024 agrees that the workaround is a workaround. Training on both orders repairs reversal tasks by doubling the data (Golovneva et al., 2024). The best autoregressive repair I found reaches 50% on reversal with a one-billion-parameter model (Ma et al., 2026a). An analysis with Google DeepMind co-authors finds that bidirectional fixes store the forward and the reverse direction as separate entries (Coda-Forno, Wang & Chaudhry, 2026), and a companion study that long reasoning at test time solves several failures of this family and "does not unlock direct knowledge inversion" (Chaudhry, Thiagarajan & Lampinen, 2026). Methods that edit the weights directly forget earlier edits as new ones accumulate (Gupta, Rao & Anumanchipalli, 2024), and when an edited fact has consequences, putting the fact in the context beats every editing method tested (Cohen et al., 2024). The remedy that wins is the one this section prices at 21 times the energy.

6 The third tax: a fixed price per token

A mathematician who has seen a problem before answers it faster, and a brain that answers faster spends less on that answer. I wanted to know whether the same is true of a transformer. On a single NVIDIA L40S, at batch size 1, I ran six Qwen2.5 instruction models from 0.5B to 32B parameters, quantized to 4 bits (Q4_K_M, llama.cpp). Each model received 20 *familiar* and 20 *unfamiliar* items of the same kinds of counting task. Familiar items have answers that appear in standard references, such as a classical permutation count or a power of two. Unfamiliar items have no published answer. I had planned the familiar items as recitation, and they were not recited. The models worked them out at length and got 0 to 13 of 20 right depending on size, against 0 to 2 of 20 on the unfamiliar ones. So the comparison is between instances of one procedure

that the model has or has not met before. It tests no contrast between recall and reasoning. I also ran 172 real user prompts from WildChat (Zhao et al., 2024), sorted into six topics, on the 7B and 14B models.

Energy is the integral of card power, sampled through NVML at 10 Hz, minus the idle baseline measured before each model. Dividing energy by tokens is unstable in short generations, where prefill and the power ramp weigh more than the tokens. So for each model and condition I fit a line, energy = $c_0 + c_1 \times$ output tokens, over every generation of at least 20 tokens, and compare the slope c_1, the energy of one more token (Table 6, Figure 8). The intercepts come out negative, because energy per token rises with length (see below), so c_0 is not a fixed cost and I do not interpret it. The analysis covers 3B to 32B. In the 0.5B model every generation that I kept runs to the 8,192-token limit, and in the 1.5B model 12 of 31 do. Those are degenerate loops ("2 * 2 * 2 …") that test very little, and I leave both models out.

Table 6. Marginal energy per output token, the slope c_1 of a fit of energy on output tokens, for familiar and unfamiliar instances. Qwen2.5 at 4-bit on one L40S, batch 1.

Model	n (familiar / unfamiliar)	c_1 familiar (J/token)	c_1 unfamiliar (J/token)	Ratio unfamiliar / familiar [95% CI]	90% CI
3B	15 / 13	0.61	0.60	0.98 [0.96, 1.02]	[0.96, 1.01]
7B	15 / 15	1.32	1.29	0.98 [0.92, 1.05]	[0.93, 1.05]
14B	16 / 15	2.65	2.60	0.98 [0.90, 1.08]	[0.91, 1.07]
32B	16 / 15	5.66	5.93	1.05 [0.96, 1.06]	[0.98, 1.06]

The intervals come from 2,000 bootstrap resamples of generations; the script and its output are in the reproduction package. Every 95% interval among the four models analysed contains one; in the excluded 0.5B model, whose generations are loops, it sits just above one, at 1.00 to 1.05. No difference larger than about 10% was detected. Dividing mean energy by mean tokens instead gives ratios up to 1.30 at 3B, and that is length again: unfamiliar generations run longer, and energy per token rises with length. Equivalence within ±5% holds at 3B, where the 90% interval lies inside the margin. At 7B, 14B and 32B, fifteen generations per cell are too few to show it. A second analysis, which I chose after seeing the data, keeps generations of at least 2 seconds and compares medians of energy per token. It gives ratios of 0.99, 0.99 and 1.01, with 95% intervals between 0.95 and 1.05. That filter excluded only familiar generations in the three largest models, 14 of them, 9 of those correct, and those are the cases closest to recall. I report it only as a sensitivity check.

Real traffic tells the same story (Figure 9). The six topics span 8% at 7B and 5% at 14B in energy per token, from cheapest to most expensive. The cheapest topic, relationships, was curated by hand, and without it the spread is 5% and 1.5%. What does move the price is model size, at about 0.18 to 0.19 J per billion parameters per token at 7B, 14B and 32B. Within a model the price also rises with output length, because attention reads a growing context (Vaswani et al., 2017). For comparison, Samsi et al. (2023) measured about 3 to 4 J per output token for a 65B model on sharded V100 and A100 GPUs.

A dense model is expected to behave this way, since every parameter takes part in every token, and the field knows it. Raposo et al. (2024) open their paper by noting that transformer language models spread

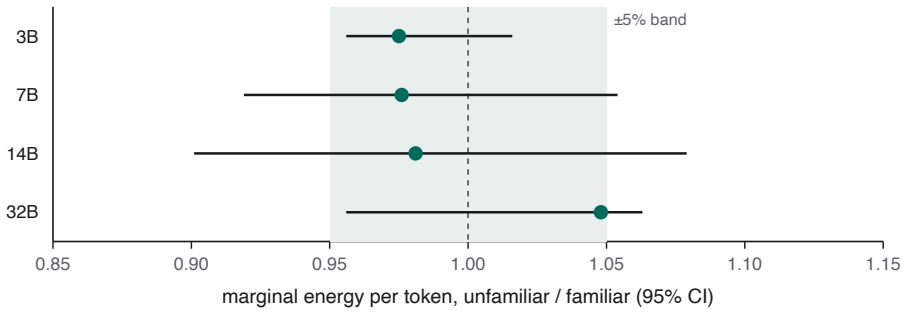

Figure 8. Marginal energy per output token, unfamiliar over familiar instances, from a fit of energy = $c_0 + c_1 \times$ tokens per model and condition over all generations of at least 20 tokens; bars are 95% bootstrap intervals, the shaded band is ±5%. No difference larger than about 10% is detected; equivalence within 5% is shown only at 3B.

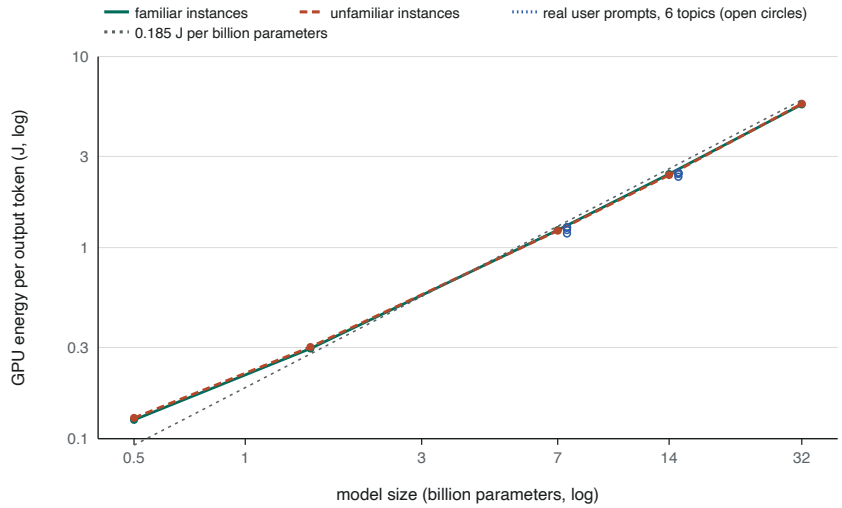

Figure 9. Energy per output token for Qwen2.5 models at 4-bit on one L40S, batch 1: median per model and condition in the post hoc sensitivity set (at least 20 tokens and 2 s; 3B omitted, too few familiar generations; 0.5B and 1.5B are loops to the token limit). Familiar and unfamiliar instances overlap at every size; open circles are six topics of real user prompts on the 7B and 14B models. The dotted line is 0.185 J per billion parameters, the mean marginal price at 7B to 32B.

compute uniformly across a sequence. A 2026 measurement paper says it in one line: "every weight is read from memory for every generated token" (Ma et al., 2026b). The statement holds for dense models, which are all I measured. Mixture-of-experts models, common at the frontier, route each token through a subset of their feed-forward parameters, and that is a form of paying for activity already in production; one direct measurement finds 12% to 18% less energy per token from routing at the level of the batch (Pina & De Silva, 2026). The router chooses by content. Whether it charges less for an easy instance is the test of this section, and I have not seen it run on such a model. My price per parameter is also a property of how full the card is as much as of the model: at batch 1 the hardware idles between tokens, and a saturated server brings the same quantity down about thirtyfold (Section 10). Adaptive computation, first in recurrent networks (Graves, 2016) and then in transformers (Dehghani et al., 2019; Raposo et al., 2024), is a line of work that exists to undo this. What this section adds is the measured number, with an interval around it.

If the price per token is fixed, the bill is the token count. The same 172 real prompts show it. On the 14B model a writing request cost 1,508 J on average and small talk 363 J, a ratio of 4.2, and the answers were 615 and 149 tokens long, a ratio of 4.1 (Figure 10). The meter runs on length. A hard subject costs more only when it makes the model talk longer.

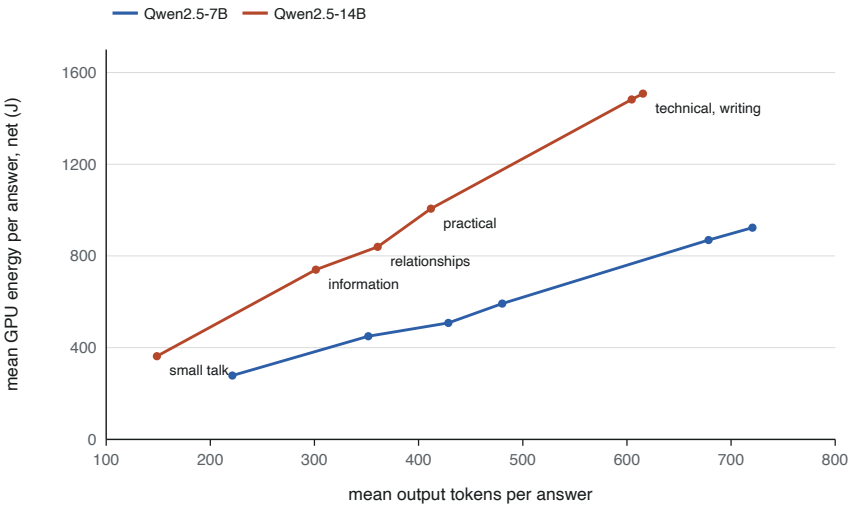

Figure 10. Mean GPU energy per answer against mean answer length for 172 real user prompts in six topics (WildChat), on two models. Within a model the points fall on a line: the cost of an answer is its length.

The token count is where novelty gets expensive. I gave four hosted frontier models the same counting problem in two forms, for board sizes n from 8 to 16: one whose answer is a classical sequence found in any reference, and one drawn at random, with the same kind of restriction, whose answer exists nowhere. All 40 answers were correct. For the classical form the models spent 4 to 200 output tokens. For the new one they spent 830 to 31,000, not always monotonically, growing about 1.3-fold for each unit of n, and at n = 16 the new form cost 64 to 121 times the tokens of the classical one (Table 7, Figure 11). I audited the reasoning on the two largest sizes: 8 of 8 correct, no tool declared, and the intermediate rook numbers the models wrote down matched an independent computation. These were hosted models, so I have tokens and no joules, and two of them hide part of their reasoning, which makes their counts a lower bound. A long derivation is work and not a tax: a person pays for it too, and my own simulation of a human solver in Section 8 has the same gap at about a hundredfold. The tax is that each of those tokens is billed at the full fixed rate, with no discount for the steps that are routine.

Table 7. Output tokens spent by four hosted frontier models on a counting problem whose answer is in the references (recite) and on a random instance of the same kind (derive), by board size n. All 40 answers correct.

Model	Recite, n = 8 to 16	Derive, n = 8 to 16	Growth per unit of n (derive)	Derive / recite at n = 16
GPT-6 Astra	46 to 104	832 to 7,249	× 1.31	70×
GPT-6.1 Sol	22 to 95	932 to 6,683	× 1.29	70×
Claude Opus 5.5	4 to 199	1,816 to 12,735	× 1.31	64×
Claude Fable 5.1	4 to 151	4,050 to 18,223 (31,292 at n = 14)	× 1.27	121×

On my own GPU the same contrast can be put in joules, with a small sample. In the one family of items where the 32B model succeeded in both forms, large powers asked in a base it has seen and in a base it has not, it got 5 of 5 familiar items for 346 J per correct answer and 2 of 5 unfamiliar items for 10,400 J per correct answer, a factor of 30 resting on two successes. In the other families it solved no unfamiliar item, and two attempts that ran to the token limit took most of the 106 kJ one family consumed. An earlier draft of mine pooled all families into one "30×". The number was similar and the arithmetic was wrong, because it divided the energy of one family's failures by another family's successes.

Doubt is bought at the same counter. In a side experiment I gave transformers a perceptual decision, 200 left and right arrows with the evidence removed, and a paid option to skip. Answering in one pass, three open-weight models skipped on 4 of 900 trials and never when there was nothing to see. A model built to

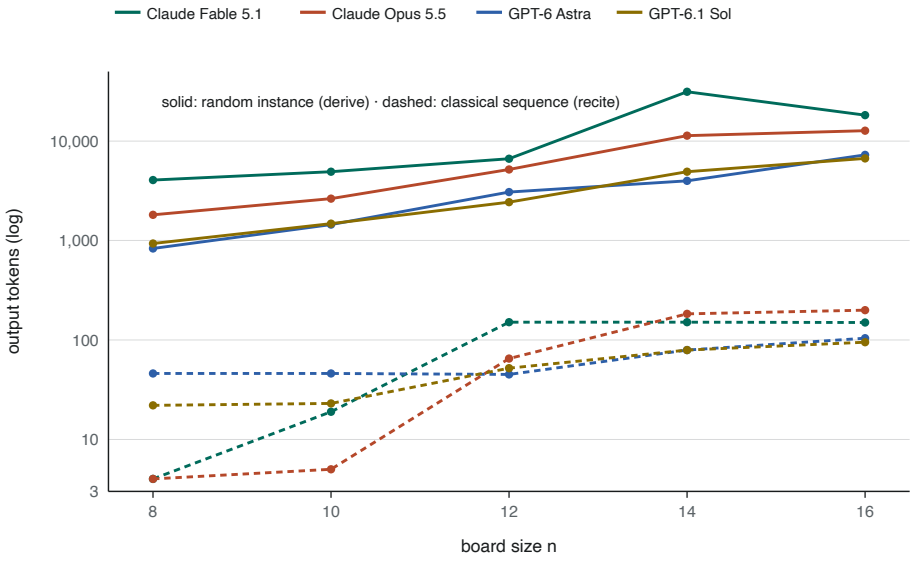

Figure 11. Output tokens spent by four hosted frontier models on a counting problem with a classical answer (dashed) and on a random instance of the same kind (solid), by board size. All 40 answers were correct. Tokens, not joules; two of the models hide part of their reasoning. Log scale.

reason got fifteen trials per condition under a cap of 12,000 tokens. It finished ten of the evidence-free trials and gave up on five of them. With strong evidence it finished nine and never gave up. Counting the trials cut at the cap, the median spend was 8,723 tokens with nothing to see and 7,844 with strong evidence (Mann-Whitney $p = 0.77$). The sample is small, and the raw trials are in the deposit. The doubt was charged whether or not there was anything to doubt. A guess from a one-pass model costs one token.

The brain's budget is also close to fixed per second. The difference is how little activity moves it, and how little of the tissue is active. It is far from free at rest, since resting potentials and housekeeping take a large share of the budget (Howarth, Gleeson & Attwell, 2012), and by the test of this section the whole organ would fail as well: its bill per second barely moves with the task. What the brain has is a low price for the quiet part. The cost of a spike limits the fraction of cortical neurons substantially active at once, possibly to fewer than 1%, with an optimal sparseness between 1% and 4% (Lennie, 2003). In rodent grey matter, signalling takes a large fraction of the energy budget (Attwell & Laughlin, 2001), and a code gets cheaper as fewer units fire (Levy & Baxter, 1996). The transformer bills every parameter on every token, and the cortex keeps most of its units quiet. Whether that makes the brain cheaper per solved task can only be settled by measuring people on the same tasks. Section 8 says what I expect that measurement to show, and it is not flattering to brains.

7 The fourth tax: agency

A person asked to fix a bug opens the file, remembers what they read a minute ago, tries something, sees that it failed and tries something else. None of this is billed separately. It comes with the 100 W.

A language model has none of it by construction. It has no state between calls, so an agent built on it re-reads its own history at every step. It cannot act, so each action is a tool call whose result is appended to the history and read again. The industry's own numbers describe the multiplier. An agent uses about 4 times the tokens of a chat and a multi-agent system about 15 times (Anthropic, 2025). On a standard software benchmark, coding agents consume on the order of a thousand times the tokens of code reasoning without tools, and two runs of the same task can differ thirtyfold (Bai et al., 2026). Turning on reasoning multiplies GPU energy by 30 on average across open models, and by 150 to 700 within single models (Luccioni & Gamazaychikov, 2025).

In 2026 someone finally put a meter on it. Luo et al. (2026) ran an open 27-billion-parameter model as a coding agent on a two-GPU server and measured about 2,989 kJ of system energy to complete one benchmark task, 830 Wh. Per output token the agent cost 63 times what the same model costs in saturated serving, because a sequential agent leaves the hardware idle between steps and spends most of its tokens reading. A climate scientist who logged eight weeks of his own use of a frontier coding agent counted 3.2 billion tokens, 96% of them re-reads of cached context, and estimated 150 Wh per prompt, about 600 times a chat prompt, and 3.0 kWh for an average day (Hausfather, 2026). His tokens are measured and his energy per token is assumed. I found no direct energy measurement of an agent task on a hosted frontier model (Table 8, Figure 12).

Table 8. The price list of agency. Measured means a power meter on the hardware; estimated means tokens counted and energy per token assumed. Human rows are power × time.

Unit of work	Energy	Basis	Source
Median chat prompt	0.24 Wh	measured, full stack	Elsworth et al. (2025)
Reasoning query, 5,000 output tokens	3.9 Wh	modelled, node and facility	Oviedo et al. (2026)
Coding-agent session, median	41 Wh	estimated from tokens	Couch (2026)
Coding-agent prompt, frontier model	150 Wh (60 to 290)	estimated from tokens	Hausfather (2026)
One benchmark software task, open 27B model	830 Wh	measured, system	Luo et al. (2026)
Brain, eight working hours	160 Wh	20 W × 8 h	—
Body, eight working hours	800 Wh	100 W × 8 h	—
Body, one full day	2,900 to 4,000 Wh	FAO requirement, adults	FAO/WHO/UNU (2004)
Coding agent, heavy user, one day	3,000 Wh	estimated from tokens	Hausfather (2026)

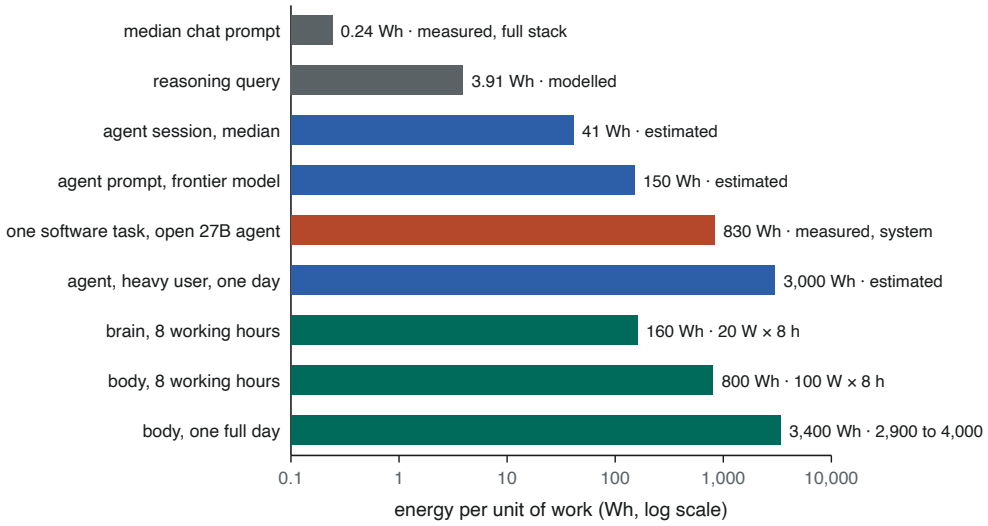

Figure 12. The price list of agency (Table 8). Grey: chat and reasoning queries. Blue: estimates from counted tokens and an assumed energy per token. Rust: the one system-level measurement, on an open model. Green: a person, as power × time. Log scale.

Table 8 mixes units that are not interchangeable, a prompt, a session, a task and a day, each with its own boundary, and only two of its machine rows come from a meter. With that warning, two lines deserve to be read together. One measured software task, on a dedicated server that sat idle between the agent's steps, cost as much as a resting human body spends in a working day, and a heavy day of one agent is estimated at what a whole person eats. On chat, the model and the brain were tied. On agency the estimates put the model at roughly a human being's energy, spent to rent what the human being came with. The comparison leaves out speed and quality in both directions: in a controlled benchmark of research-engineering tasks, agents scored four times higher than human experts given two hours, and humans given thirty-two hours scored twice as high as the best agent (Wijk et al., 2024).

Reliability compounds the bill. An agent that completes each step correctly 98% of the time completes ten steps 82% of the time and thirty-five steps 49% of the time. Every failed run is paid in full. This is why the unit proposed in Section 10 divides energy by verified successes, and why one group that measured quantized agents found that 4-bit weights, cheaper per token, could cost up to 3.5 times more energy per completed task (Abkin et al., 2026).

Where I looked for a tax and found a confound. I expected the machine to fail at finding shortcuts. The shortcut is what makes a mathematician cheap: once you have the recurrence, each new case costs a few multiplications. I designed three families of restricted permutations whose rules alternate with the parity of the position, checked that their terms a_4 to a_{13} return nothing in the OEIS, and asked models to find a recurrence and use it to produce the first fourteen values. An answer counts as correct when all fourteen values match a brute-force count. I also transcribed each stated recurrence or transfer matrix into a verification script and checked it against the brute-force count up to n = 20 (Table 9, Figure 13).

Table 9. Hosted frontier models in agent mode on three permutation families absent from the OEIS. ✓ means all fourteen values correct; the numbers are output tokens.

Hosted model (agent mode)	N1	N2	N3	Mean output tokens
Claude Opus 5.5	✓ 7,004	✓ 4,442	✓ 14,410	8,619
GPT-6 Astra	✓ 7,147	✓ 2,270	✓ 5,344	4,920
Claude Fable 5.1	✓ 8,839	✓ 5,714	✓ 13,838	9,464

Nine out of nine. I had registered one to three successes per model, each above 5,000 tokens. Accuracy landed at the top of that range, and two of the nine runs came in under 5,000 tokens. The models built transfer matrices over pairs of positions, which is how a combinatorialist would do it. All of them held to n = 20 from the index where they apply. One model stated a starting index one too early, and its values were right.

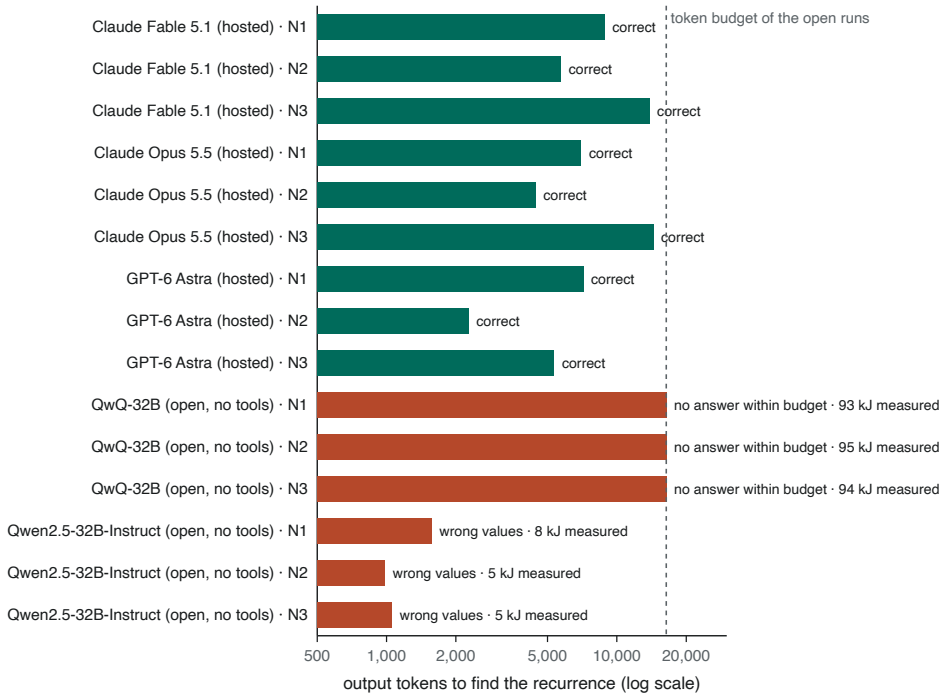

Figure 13. Output tokens spent finding a recurrence for three permutation families absent from the OEIS. Green: all fourteen values correct. Hosted models ran in a service that may expose tools; open models ran on my own GPU with no tools and metered energy. Log scale.

These models ran as tasks inside a hosted coding agent, with 15,000 to 25,000 tokens of system context, in a cloud session where tools may be available, at an unknown temperature, and one of them hides its reasoning. I then ran the same three prompts on two open 32B models on my own L40S, with no tools, temperature 0, 4-bit weights and a budget of 16,384 output tokens (Table 10).

Table 10. The same three prompts on open 32B models on my own GPU, with no tools and metered energy.

Open model, no tools	Solved	Output tokens per family	GPU energy per family, net
Qwen2.5-32B-Instruct	0 of 3	984–1,572	5.1–7.9 kJ
QwQ-32B	0 of 3	16,384, budget exhausted every time	93–95 kJ

The instruction model answered fast and from memory. For N2 it gave the powers of four, for N3 the Fibonacci numbers, and for N1 a sequence that looks geometric. None of them is the answer, and the first two are among the most familiar sequences in mathematics. Asked for something new, it handed back something famous. The reasoning model thought for nine minutes per family and hit the budget without an answer. Each attempt cost about 94 kJ, which is 78 minutes of a 20 W brain. Both outcomes match what I registered for the open models.

Between the two tables, four things changed at once: the model, the tools, the numeric precision and the token budget. Three hosted agents, possibly with tools, solved all nine. Two open 32B models without tools, at 4-bit and a 16,000-token budget, solved none. This design cannot say which of those made the difference, and I cannot put joules on the hosted successes. What it shows about agency is the size of the gap between a bare model and a model wrapped in everything a provider wraps around it, with a system prompt longer than most of the answers. The wrapping is where the fourth tax is collected, and the provider is the only one who can read that meter.

8 Where the price of the new is not yet known

Three results in my own files go against the story a neuroscientist would like to tell. They stay in.

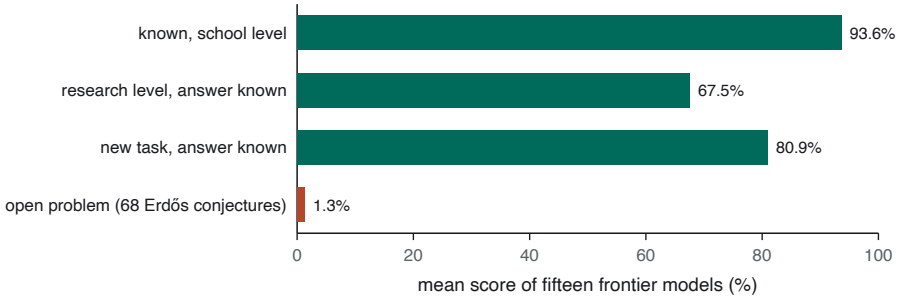

Figure 14. Mean score of fifteen frontier models by whether the answer already exists (Table 11). The grouping was made after seeing part of the scores.

The machines do solve open problems. I sorted the public benchmark scores of fifteen frontier models into four steps, a classification I made after seeing part of the scores (Table 11, Figure 14). On school-level science they average 93.6%. On research mathematics and physics whose answers are known to the examiners, 67.5%. On tasks built to be new to the model, with a known answer, 80.9%. On a benchmark of 68 Erdős conjectures that were open in August 2026, formalised in Lean and verified mechanically, they average 1.3%, and the best official run solved 3 (Epoch AI, 2026c). The drop does not follow difficulty, since the third step scores above the second. It sits between problems whose answer exists somewhere and problems whose answer does not. But seventeen of the 68 conjectures have by now been solved by machines: three in the official run, under a cap of US$ 300 and 72 hours per attempt, and the rest in further attempts by the benchmark's authors and in a developer's own publication, where no cap is stated. "The machine cannot do what is new" is false. What nobody publishes is what a success cost, counting the failures.

Table 11. Mean score of fifteen frontier models on public benchmarks, grouped by whether the answer already exists. Grouping made by me after seeing part of the scores. Data: Epoch AI benchmarking hub, November 2026.

Step	Benchmarks	Mean score	Model–benchmark pairs
Known, school level	GPQA Diamond	93.6%	14
Research level, answer known	FrontierMath tiers 1–4, CritPt, ProofBench	67.5%	59
New task, answer known	ARC-AGI-2, EnigmaEval	80.9%	17
Open problem	FrontierMath Erdős (68 conjectures)	1.3%	15

On serial counting, the machine is cheaper than the person. Before recruiting anyone I simulated people solving the same boards by hand, with assumed times per step and error rates for three profiles. It is a model with declared parameters and not a measurement, and I report it as a registered prediction. A careful solver needs about 35 minutes for n = 8 and gets it right 91% of the time, about 46 kJ of brain per correct answer at 20 W. At n = 10 the simulated person needs over three hours and is right half the time, 437 kJ per correct answer, against 4 to 14 kJ for the hosted models if their tokens are priced at 2.8 J each, the 3.9 Wh per 5,000-token reasoning query modelled by Oviedo et al. (2026); the published range for such a conversion runs from 0.4 to 14 J per token and does not change the order. At n = 12 the person almost never finishes. Halving or doubling the assumed speeds moves the numbers and keeps the order. What kills the human is the accumulation of slips over thousands of serial steps. Counting a board is the kind of work brains are bad at and should not be used to argue that brains are cheap.

The same simulation shows where the person wins. Someone who knows the closed formula for the classical case answers n = 16 in three minutes, about 4 kJ. The human advantage lies in having found the shortcut once, after which every case is nearly free. That is the comparison Section 7 tried to make and could not price. A human bench to measure it exists, with the same boards for n from 5 to 8, timed in the browser, a stated consent form and no data yet (INHS, 2026).

Two failures of 2023 are gone. On ten classic trick questions, five frontier models scored 10 of 10; on twenty reworded variants, 17 to 20, with the variants costing up to 2.3 times the tokens. And when I told four models the same facts in the voice of a hopeful, a neutral and a sceptical narrator, the narrator moved

their probability estimates by less than one percentage point, against 79 to 85 points for the evidence. They judged by the facts. I had predicted they would not.

9 Who is trying to pay less

Each tax has people working to abolish it. I went through the work of 2023 to November 2026 with one question for every claim: was it read off a meter, and on what? The literature mixes four boundaries: energy estimated by counting operations, energy measured on a chip, energy measured on a system, and dollars per task. Nearly every gain of several orders of magnitude lives in the first. Table 12 keeps the rows where I could find a number, with its boundary.

Table 12. Who is trying to pay less. Boundary: est. = estimated from operation counts or circuit models; chip, system = measured there; US$ = price. Taxes: 1 building the learner, 2 learning after deployment, 3 fixed price per token; agency is the fourth.

Line of work	Result	Boundary	Tax	What is still missing
Neuromorphic chip, Loihi 2 (Abreu et al., 2025)	370M matmul-free model on 24 chips: 405 mJ/token, against 723 mJ for a 400M transformer on an edge GPU and about 5.5 J for a 370M transformer on a data-centre GPU	chip	3	1.8× on the meter against the edge GPU; the same paper's single-chip estimate was 59 mJ
NorthPole (IBM Research, 2024; DeBole et al., 2025)	3B model at 28,356 tokens/s, 72.7× the energy efficiency of the next-lowest-latency GPU; a 30 kW rack serves three 8B instances	system	3	no joules per token in the sources I read; inference only
One-bit weights, BitNet b1.58 (Ma et al., 2024; 2025)	71.4× less arithmetic energy; 8.9× the throughput at 70B	est.; system for throughput	3	energy estimated from an operation model; the 2B model was trained on 4 trillion tokens
Spiking language models (Zhu et al., 2023; Wang, 2026)	32× fewer operations; a recurrent spiking language model cannot go below about 50% firing, a spiking transformer reaches 2% at a cost in cache memory	est.	3	no metered language model; sparsity depends on the task
Analog and thermodynamic hardware (Ambrogio et al., 2023; Jelinčič et al., 2025)	12.4 TOPS/W on speech in silicon; 10^4× less energy projected for image generation	chip; est.	3	no language model on the hardware
Adaptive depth and early exit (Raposo et al., 2024; Elhoushi et al., 2024)	same quality with fewer FLOPs per pass; up to 2.2× speed	system, speed	3	budget fixed in advance; no easy-versus-hard ratio reported
State-space and diffusion language models (Gu & Dao, 2023; Inception Labs, 2025)	5× throughput; over 1,000 tokens/s	system, speed	3	throughput is not joules; same training diet
Tiny recursive networks (Jolicoeur-Martineau, 2025)	7M parameters: 45% on ARC-AGI-1, 8% on ARC-AGI-2	quality	1, 3	one network per puzzle domain; no language
Sensorimotor learning, Monty (Leadholm et al., 2025)	about 5×10^8 times fewer training FLOPs than a pretrained vision transformer at equal accuracy; no forgetting	est. (FLOPs)	1, 2	77 simulated 3-D objects; no language
Test-time memory, Titans and Hope (Behrouz et al., 2025a; 2025b)	1.3B model: perplexity 14.39 against 17.92 for a transformer	quality	2	no energy, no reversal test; the authors say forgetting "is not 'solved' in general"
Self-editing weights, SEAL (Zweiger et al., 2025)	incorporating a passage: 33.5% → 47.0%	quality	2	sequential edits degrade earlier ones
Memory outside the weights (Eyuboglu et al., 2025; Lin et al., 2025)	38.6× less memory than full context; about 5× less compute at test time	system; compute	2	makes re-reading cheaper; the model has not learned
Data-limited training, BabyLM (Charpentier et al., 2025)	100M words: grammar on a par with a 70B model	data	1	below humans on understanding
Routing and expert mixtures (Pina & De Silva, 2026; Liu et al., 2023)	12% to 18% less energy per token with batch-level expert routing; over 2× lower latency from contextual sparsity	system	3	routing follows content; no familiar-versus-unfamiliar ratio reported

Line of work	Result	Boundary	Tax	What is still missing
Energy per completed agent task (Abkin et al., 2026)	4-bit weights save per token and can cost up to 3.5× more per completed task	system	agency	two models, one GPU type
The incumbent (Elsworth et al., 2025)	median prompt energy down 33× in a year	system	3	median only; total electricity of the same company up 37%

Three things stand out for someone choosing where to work.

The alternatives have orders of magnitude on paper and one digit on the meter. The same matmul-free network was reported at 70.8 mJ per token when composed from a single-chip estimate in its first paper (Zhu et al., 2024), at 59 mJ by the same method in the second, and at 405 mJ when 24 chips were measured (Abreu et al., 2025). That measurement still beat a transformer of similar size on an edge GPU by 1.8 times and one on a data-centre GPU at batch 1 by about 14. The 71-fold saving of one-bit weights is the arithmetic of matrix multiplication under an energy model, and the measured end-to-end gain is under tenfold. The largest neuromorphic system built, with 1.15 billion neurons, draws up to 2,600 W (Intel, 2024), about 130 brains' worth of power for a little over 1% of a brain's neurons (Azevedo et al., 2009), and I found no language model measured on it. I work in this direction and I would rather the reader hear this from me.

The incumbent is improving faster per query than any challenger has shown on a system. A 33-fold drop in a year, 23-fold of it from the model, is larger than any measured gain in Table 12. The incumbent's problem is the other column of Table 1.

Against the second tax, everything that works today works outside the weights. Trained caches, structured memories and thinking ahead of the question make re-reading cheaper by factors of 5 to 40. None of them makes the model know the fact. Whatever writes to the weights forgets, and the authors of the most advanced continual-learning architecture say so themselves.

There are also people who reject the recipe and propose another. Sutton argues that the scalable method is learning from experience, with goals, and his school's measured result is that standard deep networks lose plasticity under continual learning until they are no better than a linear model (Dohare et al., 2024). LeCun's programme predicts in a latent space instead of generating tokens; its strongest result is robot planning from under 62 hours of robot video, after pretraining on a million hours (Assran et al., 2025). Chollet's contribution is the scoreboard itself, which reports cost per task: in 2025 the best open entry reached 24% on ARC-AGI-2 at about US$ 0.20 a task, and a refinement loop around a frontier model reached 54% at about US$ 30 (Chollet et al., 2026; ARC Prize, 2025b). Hinton's mortal computation imagines trillion-parameter networks on a few watts of analog hardware and has no measurement yet (Hinton, 2022). Friston's group learns ten small games in 10,000 steps with priors that are, in their words, engineered (Heins et al., 2025). Hawkins's group is the Monty row above. Each has shown one thing the transformer does not do cheaply, and none of them has shown language. A predecessor of the tiny recursive networks in Table 12 is a caution here: an independent audit found that its celebrated hierarchy contributed little and that most of the score came from an outer refinement loop and training on the test tasks (ARC Prize, 2025).

My search was the one behind Table 12, about a hundred papers and vendor reports read for their numbers, and it is not exhaustive. Within it, I found no work, by challenger or incumbent, that reports joules per verified answer, the cost ratio of unfamiliar to familiar instances, and two-way recall after learning, for the same network. One group reports the first (Abkin et al., 2026) and the reversal literature reports the third. Section 10 asks for those three and two more.

10 A budget for doing it differently

I state requirements and budgets and no design. The tests are the part of this paper I can defend, and a number from my own laboratory without its method would be the advertisement I promised not to write.

The objective was chosen without discussion. Levy and Baxter showed in 1996 that the neural code which maximises information capacity is different from the one which maximises information per unit of energy, and that the energy-efficient code fires far fewer units. The transformer recipe optimises the first objective: capability per parameter, and then throughput per dollar. Nervous systems were selected on the second. A chemical synapse transmits a bit for about 10^4 molecules of ATP (Laughlin, de Ruyter van

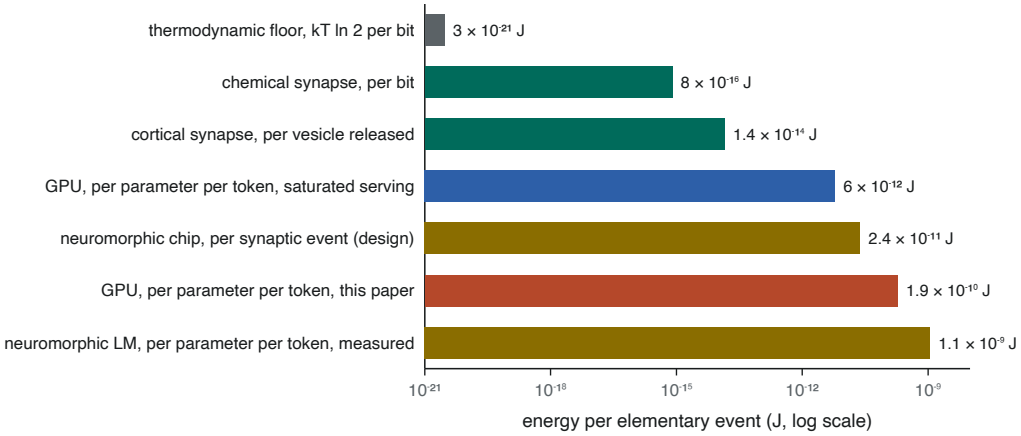

Figure 15. Energy per elementary event. Synapse figures: Laughlin et al. (1998) and Attwell & Laughlin (2001), with ATP at 50 kJ/mol. Chip design figure: Davies et al. (2018). GPU, saturated: Chung et al. (2025), 70B model at large batch. Neuromorphic language model: Abreu et al. (2025), 405 mJ per token over 370M parameters. Log scale; orders of magnitude.

Steveninck & Anderson, 1998), around 8×10^{-16} J, a figure measured in the fly's retina and five orders of magnitude above the thermodynamic floor of kT ln 2. Measuring accuracy per joule instead of accuracy changes what one builds, and it makes "I don't know" a valid output, because silence is the cheapest correct answer there is.

What one parameter costs. My measurements give a price for touching one parameter once: 0.18 to 0.19 J per billion parameters per token is about 1.9×10^{-10} J per parameter per token, on one L40S, at 4 bits, batch 1. In saturated data-centre serving the same quantity is around 6×10^{-12}, from 0.37 to 0.42 J per token for Llama 3.1 70B on four H100s at batch 512 to 900, GPU only (ML.ENERGY leaderboard data; Chung et al., 2025). Newer hardware in the same data is already at half that. A cortical synapse spends about 1.4×10^{-14} J per vesicle released, post-synaptic cost included (Attwell & Laughlin, 2001; rodent cortex, 1.64×10^5 ATP at 50 kJ/mol). A neuromorphic chip's design figure is 2.4×10^{-11} J per synaptic event (Davies et al., 2018). So one touch of a parameter costs between four hundred and fourteen thousand synaptic events, depending on how well the GPU is batched (Figure 15). The spread is the warning. Joules per parameter per token measure how full the card is as much as what the model is: the figure falls thirtyfold with batching, and for a 370M model alone on a data-centre GPU it is some eighty times worse than mine.

A cortex and a 32B model on one GPU present similar bills. The human neocortex has about 1.5×10^{14} synapses (Pakkenberg et al., 2003), some 4,700 times the parameters of the largest model I measured. Speaking at 160 words a minute on 20 W, the whole brain spends 7.5 J per word. My 32B model spends 5.7 to 5.9 J per token. The two bills are the same size, for systems five thousand times apart in connection count. The coincidence belongs to batch 1: in saturated serving the 32B model costs about 0.2 J per token and is some forty times cheaper per word than the brain. Run the comparison the other way. A dense network with as many parameters as the neocortex has synapses, billed at my measured rate and at one token per word, would spend about 28 kJ per token and need about 75 kW to keep up with speech; at the saturated data-centre rate, over 2 kW. The brain does it on 20 W while also seeing, moving, regulating a body and learning from the conversation. I am equating a synapse with a parameter here, which no neuroscientist will forgive, so read one significant figure and stop. The order says the brain's economy does not come from cheaper components alone. Most of the organ is quiet at any moment, and a quiet synapse costs much less than a busy one.

Table 13. Energy per token for a 32-billion-parameter system under five billing rules. Row 1 is measured and row 2 is a measurement on a 70B model scaled by size; rows 3 to 5 are arithmetic, fraction active × parameters × energy per event, and are floors, since they leave out everything a system spends at rest. The last row is what a meter showed for the one language model run on neuromorphic silicon.

Billing rule	Energy per event	Fraction active	Energy per token
Dense, one GPU, 4-bit, batch 1 (this paper)	1.9×10^{-10} J	100%	5.7 to 5.9 J, measured
Dense, saturated data-centre serving (Chung et al., 2025, scaled by size)	6×10^{-12} J	100%	about 0.2 J
Activity-priced, same GPU cost per event	1.9×10^{-10} J	2%	0.12 J
Activity-priced, neuromorphic design figure (Davies et al., 2018)	2.4×10^{-11} J	2%	15 mJ
Activity-priced, cortical synapse (Attwell & Laughlin, 2001)	1.4×10^{-14} J	2%	9 μJ
Measured on Loihi 2, 370M-parameter model (Abreu et al., 2025)	1.1×10^{-9} J per parameter	not reported	405 mJ for 370M parameters

Table 13 is the budget, and it should be read as a set of floors. Its fourth row says that a 32B system which paid only for 2% activity at the chip's design figure would spend 15 mJ per token, about 400 times less than I measured. The rule that produces that number fails on the one system where it can be checked. Applied to the cortex, 2% of 1.5×10^{14} synapses at one vesicle each comes to about 40 mJ per word, and the cortex's actual bill is 7.5 J, some 180 times more, because rest, housekeeping and everything that is not the word are on it. The last row is the second check. The one language model actually run on neuromorphic silicon spent 405 mJ per token for 370M parameters, about forty-five times its own floor at full activity. Anyone can multiply by 2%. Getting a meter to print the same figure is the part that has not been done at scale, and a 2026 study adds that a recurrent spiking language model's activity could not be pushed below about 50% without losing the task, while a small spiking transformer reached 2% at the price of a full attention cache (Wang, 2026). The 2% of Table 13 is the cortex's number (Lennie, 2003). I know of no language model at scale shown to read at that fraction with a meter attached.

Five requirements, each with its test.

Pay for activity. Energy per token must fall when the instance is easy or familiar. The test is Section 6: fit energy on output tokens for familiar and unfamiliar instances of one task and report the ratio of slopes with its interval. A system that meets the requirement shows a ratio clearly above one. Every dense model I measured has an interval that contains one. Adaptive computation and expert routing move in this direction, and the same test can score them. The brain itself would pass only in part: its marginal cost follows activity and its baseline does not.

Keep a fast store beside the slow one. After learning N new facts stated in one direction, the system must recall them in both, keep its score on an old benchmark within noise, and spend less energy than holding the facts in context for the queries that will use them. The test is Section 5. The model I measured passes the energy criterion and fails recall, at 33.5% forward and 0% reverse.

Learn from exposure. Report the language and the energy consumed to reach a capability next to the capability, in the same table. Chollet (2019) proposed measuring intelligence as skill-acquisition efficiency; this requirement asks only for the bookkeeping that makes such a measure possible.

Arrive with the doubt. When evidence is removed and declining pays, the system must decline, and its cost of declining must be lower than its cost of answering. The test is the skip experiment of Section 6: tokens, or joules, spent with and without evidence, and the rate of declining in each. A brain reaches the refusal inside the same pass that would have produced the answer. In circuits I built from textbook parts, a sparse code at the honeybee's measured level declined with better judgment than a dense one that used 38 times more spikes, and pruning connections down to the fraction the task had already made redundant removed 23% to 35% of the signalling energy at unchanged accuracy. Those are unpublished spiking simulations of a decision, a long way from a language model, and the pruning result is close to a counting identity. I cite them as the reason I think the requirement can be met, and as nothing more.

Charge per verified answer. Report results on reasoning benchmarks as joules per verified correct answer, with all failed attempts in the numerator, the energy boundary stated (card, system or facility) and the verifier

named. For systems that may decline, add the risk–coverage curve, so that silence is neither rewarded for free nor punished as error. Section 7 shows why the unit matters. A model that fails at 94 kJ and a model that fails at 5 kJ have the same accuracy and very different bills. A hosted model that succeeds at an unknown cost cannot be compared with either of them. One group has already measured energy per completed agent task (Abkin et al., 2026), and the ARC Prize publishes dollars per task. So the unit is already in print. I am asking for it to become a column that every leaderboard is embarrassed to leave empty.

What the next four years cost if nothing changes. These are ceilings stacked on projections, and I give them as such. If the largest training run reaches the 4 to 16 GW that Epoch projects for 2030 and lasts ninety days at full draw, as the largest disclosed run of 2025 is listed, it will consume 9 to 35 TWh, the food energy of half a million to two million human upbringings for one model that will be replaced within a year (my arithmetic on You & Owen, 2025). If token volume keeps growing even at the slower of the two rates Google reported, sevenfold a year, energy per token has to fall sevenfold a year for the total to stand still. One company reported a 33-fold fall per prompt in one year, most of it from changes to the model, and hardware contributes about 40% a year; whether software can repeat that every year is the open question, and a token is not a fixed quantity of energy. Agency moves the other way, at an estimated hundreds of times the energy of a chat prompt for each instruction. I did not have to assume that anything goes wrong to get these numbers. They are what success looks like on a meter.

A different recipe would be recognised by the bill it presents: a price that falls on easy instances, learning that does not have to be re-read or re-trained, a refusal that costs less than an answer, and a statement of joules per verified answer at the bottom. Until the invoice comes stapled to the answer, all any of us has seen is a quote.

11 Limits

- GPU energy only in my own measurements: no CPU, memory, network or data-centre overhead (PUE). This understates the machine's energy. Figures from other sources carry their own boundaries, stated where used, and are not interchangeable.
- The per-token measurements use one model family, 4-bit weights, batch size 1 and one GPU type. Batching lowers absolute J/token. It does not create a difference between conditions where there is none.
- I measured dense models only. Mixture-of-experts models do not use every parameter on every token, and the third tax as stated applies to them in a weaker form that I did not measure.
- The main analysis in Section 6 uses every generation of at least 20 tokens. Its linear fit has negative intercepts, so it is a local description of the slope and not a cost model. A post hoc 2-second rule is reported as a sensitivity check, and it excludes only familiar generations.
- The recite-versus-derive curve of Section 6 comes from hosted models: tokens and no joules, hidden reasoning in two of four, five board sizes, and a task with exploitable structure, so the growth reflects the length of a derivation and not brute force. The per-family energy ratio on my GPU rests on two successes.
- Section 5 is a single run with one seed and one configuration, and its invented names are too similar. MMLU results were stored as totals, so the change cannot be tested per question. The context and no-context routes ran at different batch sizes.
- Brain energy is an estimate of supply, not a measurement of a person at work. No human was measured on these tasks. The human simulation of Section 8 is a model with assumed parameters.
- A formation counts food energy only, from FAO requirement tables summed by me, and assumes the value for ages 17 to 18 holds to 20. The reach figures in Table 3 mix different services, and the model's reach is borrowed from one product.
- Hosted frontier models report tokens, not joules, and ran as agents with possible tools. The agent energies in Table 8 are one system measurement on an open model and several estimates from token counts.
- Children's word counts vary tenfold, and tokens are not words. The per-word comparison of Section 3 depends on an assumed reply length.

- None of the training energies in Section 4 is a developer's own measurement. GPT-3 and Grok 4 are third-party estimates, and Llama 3.1 is a ceiling from TDP. The yearly series in Section 2 covers only models that disclose power and duration, 153 of 1,092.
- The 2025 data-centre totals are the IEA's as reported by a secondary source; I could not open the IEA's 2026 report directly. Several corporate figures circulate only in the press and were left out.
- The grouping of benchmarks in Section 8 was made after seeing part of the scores.
- The six WildChat topics were assigned by an automatic classifier that agreed with a manual check on 75% of a sample.
- The synapse-to-parameter comparisons of Section 10 use rodent and insect energetics and human synapse counts, and equate a synaptic event with one use of a parameter. They are orders of magnitude.
- The skip experiment of Section 6 rests on fifteen reasoning-model trials per condition, of which ten and nine finished under the token cap, and the spiking simulations mentioned in Section 10 are unpublished and reported without their method.
- The survey behind Section 9 is a reading of about a hundred sources for their numbers and is not a systematic review. Several rows were checked at the level of the abstract only.

12 Reproduction

The items with their SHA-256 hashes, the prediction file registered before each run, the generation and analysis scripts, the raw outputs, the NVML power traces, the script that derives the training series from Epoch AI's public database, and the AWS CloudTrail records with per-instance SHA256SUMS for every GPU run are deposited at Zenodo, doi:10.5281/zenodo.24204647.

Competing interests: I direct a research institute and founded a company that builds language models, including work on designs aimed at the tests of Section 10. No result from those systems is reported here.

Sources

Abkin, S., Yang, A., Flesher, A., Kodarapu, N., Panigrahy, D., Tyagi, A. & Houngninou, D. K. (2026). Measuring the Energy Cost of Completing Tasks with Quantized Language-Model Agents. arXiv:2610.32417.

Abreu, S., Shrestha, S. B., Zhu, R.-J. & Eshraghian, J. (2025). Neuromorphic Principles for Efficient Large Language Models on Intel Loihi 2. ICLR Workshop SCOPE. arXiv:2503.18002.

Altschuler, J., Margolius, D., Bodenheimer, T. & Grumbach, K. (2012). Estimating a reasonable patient panel size for primary care physicians with team-based task delegation. *Ann Fam Med* 10(5), 396–400.

Ambrogio, S. et al. (2023). An analog-AI chip for energy-efficient speech recognition and transcription. *Nature* 620, 768–775.

Anthropic (2025). How we built our multi-agent research system. Engineering blog, 13 June 2025.

Anthropic (2026). Prompt caching. API documentation, read 13 November 2026.

ARC Prize (2025). The Hidden Drivers of HRM's Performance on ARC-AGI. Blog, 15 August 2025.

ARC Prize (2025b). ARC Prize 2025 Results and Analysis. Blog, 5 December 2025.

Assran, M. et al. (2025). V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning. arXiv:2506.09985.

Attwell, D. & Laughlin, S. B. (2001). An energy budget for signaling in the grey matter of the brain. *J Cereb Blood Flow Metab* 21(10), 1133–1145.

Azevedo, F. A. C. et al. (2009). Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain. *J Comp Neurol* 513(5), 532–541.

Bai, L., Huang, Z., Wang, X., Sun, J., Mihalcea, R., Brynjolfsson, E., Pentland, A. & Pei, J. (2026). How Do AI Agents Spend Your Money? Analyzing and Predicting Token Consumption in Agentic Coding Tasks. arXiv:2604.22750.

Balasubramanian, V. (2021). Brain power. *PNAS* 118(32), e2107022118.

Behrouz, A., Razaviyayn, M., Zhong, P. & Mirrokni, V. (2025b). Nested Learning: The Illusion of Deep Learning Architectures. NeurIPS. arXiv:2512.24695.

Behrouz, A., Zhong, P. & Mirrokni, V. (2025a). Titans: Learning to Memorize at Test Time. arXiv:2501.00663.

Berglund, L., Tong, M., Kaufmann, M., Balesni, M., Stickland, A. C., Korbak, T. & Evans, O. (2024). The Reversal Curse: LLMs trained on "A is B" fail to learn "B is A". ICLR. arXiv:2309.12288.

Brown, T. et al. (2020). Language Models are Few-Shot Learners. NeurIPS. arXiv:2005.14165.

Central Statistics Office (2026). Data Centres Metered Electricity Consumption 2025. Release of 7 July 2026.

Charpentier, L. et al. (2025). Findings of the Third BabyLM Challenge: Accelerating Language Modeling Research with Cognitively Plausible Data. Proc. 1st BabyLM Workshop, 399–420.

Chatterji, A. et al. (2025). How People Use ChatGPT. NBER Working Paper 34255.

Chaudhry, A., Thiagarajan, S. & Lampinen, A. (2026). Improving Latent Generalization Using Test-time Compute. arXiv:2604.01430.

Chollet, F. (2019). On the Measure of Intelligence. arXiv:1911.01547.

Chollet, F., Knoop, M., Kamradt, G. & Landers, B. (2026). ARC Prize 2025: Technical Report. arXiv:2601.10904.

Chung, J.-W. et al. (2025). The ML.ENERGY Benchmark: Toward Automated Inference Energy Measurement and Optimization. NeurIPS Datasets and Benchmarks. arXiv:2505.06371.

Chung, J.-W., Wu, R., Ma, J. J. & Chowdhury, M. (2026). Where Do the Joules Go? Diagnosing Inference Energy Consumption. arXiv:2601.22076.

Coda-Forno, J., Wang, J. X. & Chaudhry, A. (2026). The Illusion of Latent Generalization: Bi-directionality and the Reversal Curse. ICLR Workshop Re-Align. arXiv:2604.04943.

Cohen, R., Biran, E., Yoran, O., Globerson, A. & Geva, M. (2024). Evaluating the Ripple Effects of Knowledge Editing in Language Models. *TACL* 12, 283–298. arXiv:2307.12976.

Couch, S. P. (2026). Electricity use of AI coding agents. Blog, 20 January 2026.

Davies, M. et al. (2018). Loihi: A Neuromorphic Manycore Processor with On-Chip Learning. *IEEE Micro* 38(1), 82–99.

DeBole, M. et al. (2025). A Scalable NorthPole System with End-to-End Vertical Integration for Low-Latency and Energy-Efficient LLM Inference. arXiv:2511.15950.

DeepLearning.AI (2026). Machine Learning Specialization, course page.

DeepSeek-AI (2024). DeepSeek-V3 Technical Report. arXiv:2412.19437.

Dehghani, M., Gouws, S., Vinyals, O., Uszkoreit, J. & Kaiser, Ł. (2019). Universal Transformers. ICLR. arXiv:1807.03819.

Dohare, S., Hernandez-Garcia, J. F., Lan, Q., Rahman, P., Mahmood, A. R. & Sutton, R. S. (2024). Loss of plasticity in deep continual learning. *Nature* 632, 768–774.

Elhoushi, M. et al. (2024). LayerSkip: Enabling Early Exit Inference and Self-Speculative Decoding. ACL. arXiv:2404.16710.

Elsworth, C. et al. (2025). Measuring the environmental impact of delivering AI at Google Scale. arXiv:2508.15734.

Epoch AI (2025). What did it take to train Grok 4? Data Insight.

Epoch AI (2026a). Data on AI Models. Database downloaded 12 November 2026, CC-BY 4.0.

Epoch AI (2026b). Data on AI Data Centers. Page read 13 November 2026.

Epoch AI (2026c). FrontierMath: Erdős. Benchmark page read 12 November 2026.

Eyuboglu, S. et al. (2025). Cartridges: Lightweight and general-purpose long context representations via self-study. arXiv:2506.06266.

FAO/WHO/UNU (2004). Human energy requirements. Report of a Joint Expert Consultation. FAO Food and Nutrition Technical Report Series 1.

Frank, M. C. (2023). Bridging the data gap between children and large language models. *Trends Cogn Sci* 27(11), 990–992.

Gilkerson, J. et al. (2017). Mapping the early language environment using all-day recordings and automated analysis. *Am J Speech-Language Pathology* 26(2), 248–265.

Golovneva, O., Allen-Zhu, Z., Weston, J. & Sukhbaatar, S. (2024). Reverse Training to Nurse the Reversal Curse. arXiv:2403.13799.

Google (2026). 2026 Environmental Report. Announced in: Brandt, K., Read our 11th annual Environmental Report. Company blog, 30 June 2026.

Graves, A. (2016). Adaptive Computation Time for Recurrent Neural Networks. arXiv:1603.08983.

Greve, A., Cooper, E. & Henson, R. N. (2014). No evidence that "fast-mapping" benefits novel learning in healthy older adults. *Neuropsychologia* 60, 52–59.

Gu, A. & Dao, T. (2023). Mamba: Linear-Time Sequence Modeling with Selective State Spaces. arXiv:2312.00752.

Gundlach, H., Lynch, J., Mertens, M. & Thompson, N. (2025). The Price of Progress: Price Performance and the Future of AI. arXiv:2511.23455.

Gupta, A., Rao, A. & Anumanchipalli, G. (2024). Model Editing at Scale leads to Gradual and Catastrophic Forgetting. ACL Findings. arXiv:2401.07453.

Gurunandan, K., Cooper, E., Tibon, R., Henson, R. N. & Greve, A. (2023). No evidence of fast mapping in healthy adults using an implicit memory measure: failures to replicate the lexical competition results of Coutanche and Thompson-Schill (2014). *Memory* 31(10), 1320–1339.

Hausfather, Z. (2026). The real energy use of agentic AI. The Climate Brink, 5 August 2026.

Heins, C. et al. (2025). AXIOM: Learning to Play Games in Minutes with Expanding Object-Centric Models. arXiv:2505.24784.

Hinton, G. (2022). The Forward-Forward Algorithm: Some Preliminary Investigations. arXiv:2212.13345.

Howarth, C., Gleeson, P. & Attwell, D. (2012). Updated energy budgets for neural computation in the neocortex and cerebellum. *J Cereb Blood Flow Metab* 32(7), 1222–1232.

Hu, M. Y. et al. (2024). Findings of the Second BabyLM Challenge: Sample-Efficient Pretraining on Developmentally Plausible Corpora. arXiv:2412.05149.

IBM Research (2024). IBM's NorthPole achieves new speed and efficiency milestones. Research blog, 26 September 2024.

IEA (2025). Energy and AI. International Energy Agency, Paris.

Inception Labs (2025). Mercury: Ultra-Fast Language Models Based on Diffusion. arXiv:2506.17298.

INHS (2026). Human bench: protocol and consent form, version 1. Institute of Neuro-Humanities & Science.

Intel (2024). Intel Builds World's Largest Neuromorphic System to Enable More Sustainable AI. Press release, 17 April 2024.

Jelinčič, A. et al. (2025). An efficient probabilistic hardware architecture for diffusion-like models. arXiv:2510.23972.

Jolicoeur-Martineau, A. (2025). Less is More: Recursive Reasoning with Tiny Networks. arXiv:2510.04871.

Kahana, M. J. (2002). Associative symmetry and memory theory. *Memory & Cognition* 30(6), 823–840.

Kirkpatrick, J. et al. (2017). Overcoming catastrophic forgetting in neural networks. *PNAS* 114(13), 3521–3526.

Klinzing, J. G., Niethard, N. & Born, J. (2019). Mechanisms of systems memory consolidation during sleep. *Nat Neurosci* 22(10), 1598–1610.

Koomey, J., Schmidt, Z., Sreedharan, P., Kumar, N. & McNair, T. (2026). Separating fact from fiction in data center electricity forecasts: A guide for regulators. Koomey Analytics and GridLab, 23 March 2026.

Korinek, A. & Vipra, J. (2025). Concentrating Intelligence: Scaling and Market Structure in Artificial Intelligence. *Economic Policy* 40(121), 225–256.

Kumaran, D., Hassabis, D. & McClelland, J. L. (2016). What Learning Systems do Intelligent Agents Need? *Trends Cogn Sci* 20(7), 512–534.

Kuzawa, C. W. et al. (2014). Metabolic costs and evolutionary implications of human brain development. *PNAS* 111(36), 13010–13015.

Laughlin, S. B., de Ruyter van Steveninck, R. R. & Anderson, J. C. (1998). The metabolic cost of neural information. *Nat Neurosci* 1(1), 36–41.

Leadholm, N., Clay, V., Knudstrup, S., Lee, H. & Hawkins, J. (2025). Thousand-Brains Systems: Sensorimotor Intelligence for Rapid, Robust Learning and Inference. arXiv:2507.04494.

LeCun, Y. (2024). Interview, Lex Fridman Podcast 416, published 7 March 2024; transcript at lexfridman.com.

Lennie, P. (2003). The cost of cortical computation. *Current Biology* 13(6), 493–497.

Levy, W. B. & Baxter, R. A. (1996). Energy efficient neural codes. *Neural Computation* 8(3), 531–543.

Levy, W. B. & Calvert, V. G. (2021). Communication consumes 35 times more energy than computation in the human cortex, but both costs are needed to predict synapse number. *PNAS* 118(18), e2008173118.

Lin, K. et al. (2025). Sleep-time Compute: Beyond Inference Scaling at Test-time. arXiv:2504.13171.

Liu, P., Huang, H., Rollins, N., Chalak, L. F., Jeon, T., Halovanic, C. & Lu, H. (2014). Quantitative assessment of global cerebral metabolic rate of oxygen (CMRO2) in neonates using MRI. *NMR Biomed* 27(3), 332–340.

Liu, Z. et al. (2023). Deja Vu: Contextual Sparsity for Efficient LLMs at Inference Time. ICML. arXiv:2310.17157.

Llama Team, Meta (2024). The Llama 3 Herd of Models. arXiv:2407.21783 Llama 3.1 Model Card.

Luccioni, A. S. & Gamazaychikov, B. (2025). AI Energy Score v2: Refreshed Leaderboard, now with Reasoning. Hugging Face blog, 4 December 2025.

Luccioni, A. S., Jernite, Y. & Strubell, E. (2024). Power Hungry Processing: Watts Driving the Cost of AI Deployment? ACM FAccT. arXiv:2311.16863.

Luccioni, A. S., Strubell, E. & Crawford, K. (2025). From Efficiency Gains to Rebound Effects: The Problem of Jevons' Paradox in AI's Polarized Environmental Debate. ACM FAccT. arXiv:2501.16548.

Luo, Q., Li, K., Wang, Z. & Chen, Y. (2026). Where Does the Energy Go? Profiling LLM Agent Inference on Blackwell GPUs. arXiv:2609.29707.

Luo, Y. et al. (2023). An Empirical Study of Catastrophic Forgetting in Large Language Models During Continual Fine-tuning. arXiv:2308.08747.

Ma, B., Afzal, A., Eitzinger, J., Koestler, H. & Wellein, G. (2026b). Only Accepted Tokens Count: An Energy Accounting of Speculative Decoding Under Latency Targets. arXiv:2610.40287.

Ma, S. et al. (2024). The Era of 1-bit LLMs: All Large Language Models are in 1.58 Bits. arXiv:2402.17764.

Ma, S. et al. (2025). BitNet b1.58 2B4T Technical Report. arXiv:2504.12285.

Ma, X., Huang, Y., Zhu, H. & Sojoudi, S. (2026a). Breaking the Reversal Curse in Autoregressive Language Models via Identity Bridge. arXiv:2602.02470.

McClelland, J. L., McNaughton, B. L. & O'Reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex. *Psychological Review* 102(3), 419–457.

McCloskey, M. & Cohen, N. J. (1989). Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. *Psychology of Learning and Motivation* 24, 109–165.

Microsoft (2026). 2026 Environmental Data Fact Sheet.

Monitoring Analytics (2026). Analysis of the 2028/2029 RPM Base Residual Auction, Part A. Independent Market Monitor for PJM, 30 October 2026.

OpenRouter (2026). The Token State of Play. 22 September 2026.

Oviedo, F., Kazhamiaka, F., Choukse, E., Kim, A., Luers, A., Nakagawa, M., Bianchini, R. & Lavista Ferres, J. M. (2026). Energy use of AI inference, efficiency pathways, and test-time scaling. *Joule* 10(8), 102430. arXiv:2509.20241.

Pakkenberg, B. et al. (2003). Aging and the human neocortex. *Exp Gerontol* 38(1–2), 95–99.

Patterson, D. et al. (2021). Carbon Emissions and Large Neural Network Training. arXiv:2104.10350.

Pauliuk, S. (2024). Flawed 'LCA' study on AI gives false impression about the state-of-the-art in environmental assessment. Blog, International Society for Industrial Ecology, 30 April 2024.

Pichai, S. (2026). I/O 2026: Welcome to the agentic Gemini era. Google company blog, 19 May 2026.

Pilz, K. F., Rahman, R., Sanders, J. & Heim, L. (2025). Trends in AI supercomputers. Epoch AI, 23 April 2025.

Pina, R. & De Silva, V. (2026). Direct GPU Energy Measurements of Batch-Level Sharing in Mixture-of-Experts and Recurrent Models. arXiv:2611.11696.

Qwen Team (2024). Qwen2.5 Technical Report. arXiv:2412.15115.

Raichle, M. E. (2015). The restless brain: how intrinsic activity organizes brain function. *Phil Trans R Soc B* 370, 20140172.

Raposo, D. et al. (2024). Mixture-of-Depths: Dynamically allocating compute in transformer-based language models. arXiv:2404.02258.

Ritchie, H. (2026). How much energy do data centers and artificial intelligence use? Our World in Data, 20 July 2026.

Romer, P. M. (1990). Endogenous Technological Change. *J Polit Econ* 98(5, part 2), S71–S102.

Samsi, S. et al. (2023). From Words to Watts: Benchmarking the Energy Costs of Large Language Model Inference. IEEE High Performance Extreme Computing Conference (HPEC), 1–9. arXiv:2310.03003.

Sharon, T., Moscovitch, M. & Gilboa, A. (2011). Rapid neocortical acquisition of long-term arbitrary associations independent of the hippocampus. *PNAS* 108(3), 1146–1151.

Shehabi, A. et al. (2024). 2024 United States Data Center Energy Usage Report. Lawrence Berkeley National Laboratory, LBNL-2001637.

Sokoloff, L., Mangold, R., Wechsler, R. L., Kennedy, C. & Kety, S. S. (1955). The effect of mental arithmetic on cerebral circulation and metabolism. *J Clin Invest* 34(7), 1101–1108.

Spelke, E. S. & Kinzler, K. D. (2007). Core knowledge. *Dev Sci* 10(1), 89–96.

Tomlinson, B., Black, R. W., Patterson, D. J. & Torrance, A. W. (2024). The carbon emissions of writing and illustrating are lower for AI than for humans. *Sci Rep* 14, 3732.

United Nations, Department of Economic and Social Affairs, Population Division (2024). World Population Prospects 2024: Summary of Results.

Vaswani, A. et al. (2017). Attention Is All You Need. NeurIPS. arXiv:1706.03762.

Wang, Z. (2026). The Sparsity Ceiling: Where Spiking Networks Can and Cannot Trade Activity for Energy. arXiv:2607.26648.

Warstadt, A. et al. (2023). Findings of the BabyLM Challenge: Sample-Efficient Pretraining on Developmentally Plausible Corpora. Proc. BabyLM Challenge at CoNLL 2023, 1–34.

Wijk, H. et al. (2024). RE-Bench: Evaluating frontier AI R&D capabilities of language model agents against human experts. arXiv:2411.15114.

Wizarding World (2018). 500 million Harry Potter books have now been sold worldwide. News release, 1 February 2018.

World Bank (2018). Pupil-teacher ratio, primary (UNESCO Institute for Statistics), indicator SE.PRM.ENRL.TC.ZS.

You, J. & Owen, D. (2025). How much power will frontier AI training demand in 2030? Epoch AI, 11 August 2025.

Zador, A. M. (2019). A critique of pure learning and what artificial neural networks can learn from animal brains. *Nat Commun* 10, 3770.

Zhao, W., Ren, X., Hessel, J., Cardie, C., Choi, Y. & Deng, Y. (2024). WildChat: 1M ChatGPT Interaction Logs in the Wild. ICLR. arXiv:2405.01470.

Zheng, J. & Meister, M. (2025). The unbearable slowness of being: Why do we live at 10 bits/s? *Neuron* 113(2), 192–204.

Zhu, R.-J. et al. (2024). Scalable MatMul-free Language Modeling. arXiv:2406.02528.

Zhu, R.-J., Zhao, Q., Li, G. & Eshraghian, J. (2023). SpikeGPT: Generative Pre-trained Language Model with Spiking Neural Networks. *TMLR*. arXiv:2302.13939.

Zweiger, A. et al. (2025). Self-Adapting Language Models. arXiv:2506.10943.