PRODUCTION MEASUREMENT

FrostyTools AI text generation: a measured workload and its operational electricity cost

FrostyTools is an AI chatbot for Twitch streamers. It writes the welcomes, shout-outs, celebrations and chat replies that run during a broadcast, and also does trivia, clip finding and overlays. This paper measures the AI text generation and publishes the supporting data.

We measured **more than a million generations**, token by token, retries and failures included, against a three-hour broadcast to 200 viewers.

By **Sebastian Rapport** @rapporian

Current version: frostytools.com/research/ai-energy-use

SAMPLE	BOUNDARY	METHOD
1M+ delivered generations	Operational electricity, in use	Workload × published rate

WHY WE PUBLISHED THIS

"How much does your AI actually use?" is a fair question, and it is usually answered with silence, or with a single number and no way to check it. So we are showing our own in full: every figure, its source, and the parts that do not flatter us. We counted it because **we cannot reduce what we have not measured**, and the aim is less energy per generation without a lesser product. A small number is not an environmental credential; it is a measurement, and a place to start.

1 FINDINGS

ONE CHAT RECAP

0.05 Wh

About **one second** of a 200 W games console. With channel summaries, this is more than half of everything we run by volume, yet only 17% of our AI electricity.

A TWELVE-GENERATION SESSION

1.9 Wh

A games console for about **35 seconds**. Comparable to a single AI-assisted web search.

THE STREAM IT RUNS DURING — 3 H, 200 VIEWERS

49 kWh

End to end, every device included. **1.7 days** of a whole household's electricity, at 10,500 kWh a year.

> ### A stream uses more electricity in half a second than a FrostyTools session uses in three hours.
>
> A broadcast on this scale draws about **4.5 watt-hours every second** across 200 viewers' screens and the networks reaching them. A full twelve-generation session costs 1.9 watt-hours. Take the most conservative streaming assumption in this paper, a phone-heavy audience at 16 kWh, and the answer is still under two seconds. We put it this way rather than as a ratio because ratios between quantities of such different size invite false precision.

The gap is structural rather than incidental, and it is worth understanding rather than taking on trust. Streaming video pushes millions of pixels per second to hundreds of screens for hours on end, and **roughly seven tenths of that electricity is spent on viewers' own screens**, with most of the remainder on the networks reaching them. Generating a few hundred words of text happens once, in a data centre, and then it is over. These are activities of a genuinely different order, and no accounting method brings them close together.

Across everything FrostyTools delivered in the window, the volume-weighted mean was **0.16 watt-hours per generation**. Our two most-used features, chat recaps and channel summaries, are together more than half of everything we run by volume (yet only 17% of the electricity we use), and they cost about **0.05 watt-hours** each: comparable to one conventional web search, and roughly one thirtieth of an AI-assisted one.

2 WHAT A THREE-HOUR STREAM TO 200 VIEWERS COSTS

We start with the comparator, because it is the figure almost nobody has seen calculated. One broadcaster, three hours, 200 concurrent viewers. The accounting covers the streamer's computer and GPU, capture hardware and displays, their router, live ingest and transcoding, origin and CDN servers, the core and access networks carrying the stream, every viewer's home router, and every viewer's screen.

COMPONENT	ELECTRICITY	BASIS
200 viewers' screens televisions, laptops, phones, tablets	**33.3 kWh** 69%	The IEA measures viewing devices at 72% of streaming energy; the ITU puts a television near 100 W against under 3 W for a phone [1][2]
Networks carrying the stream home routers, access, core transit	**10.6 kWh** 22%	23% of streaming energy per the IEA; the ITU reports about 18 W for end-to-end delivery of one high-quality stream [1][2]
Data centres and CDN origin, edge cache, service overhead	**2.3 kWh** 5%	5% of streaming energy per the IEA [1]
Streamer's own setup gaming PC, GPU, capture, displays	**1.65 kWh** 3%	Full-system allowance over three hours, anchored to a desktop gaming GPU's rated power [8]
Live ingest and transcoding one input encoded to many bitrates	**0.55 kWh** 1%	Allowance for adaptive-bitrate transcoding on the source path [3]
Total	**≈ 49 kWh**	**245 Wh per viewer** across the three hours. The components above sum to 48.4 kWh; rounding to 49 slightly favours the streaming side of every comparison here, and is flagged rather than silently corrected. A television-heavy audience pushes the total toward 74 kWh; a phone-heavy one brings it down toward 16 kWh.

FIGURE 1 · WHERE 49 KWH GOES

Nine tenths of a stream's electricity is spent in the audience's homes and on the networks reaching them.

33.3 kWh 69% Viewers' screens

10.6 kWh 22% Networks

2.3 kWh 5% Data centres & CDN

1.65 kWh 3% Streamer's setup

0.55 kWh 1% Ingest & transcode

The first three segments apply the IEA's published 72 / 23 / 5 split of streaming energy to a 46.2 kWh base [1]; the two amber segments are modelled allowances for the source path. All the FrostyTools generations in a three-hour session together come to **1.9 Wh**: four thousandths of one percent of this bar, too small to draw at any width this page can print.

THE RANGE ON THIS NUMBER, AND WHY IT DOES NOT CHANGE THE ANSWER

Because viewers' own screens dominate the total, it depends heavily on what the audience is watching on. A phone-heavy audience puts the stream near **16 kWh**; a television-heavy one near **74 kWh**. We use 49 kWh as the reference case because it sits on the IEA's published intensity figure, and we carry the range wherever the number appears. The anchor is also drawn from on-demand viewing rather than live, and live audiences skew more mobile, which pushes the realistic figure toward the lower half of that range. It is contested in the other direction too: The Shift Project, whose higher estimate the IEA analysis revised downward, disputes that revision [13].

Take the most unfavourable end of every assumption. A phone-heavy audience at 16 kWh, one viewer's share at 80 Wh: a twelve-generation session is **still less than one fortieth of** than one person's share of watching, and the broadcast as a whole is still more than eight thousand times larger than the FrostyTools work supporting it. The conclusion is not sensitive to the assumption.

3 WHAT FROSTYTOOLS COSTS, FEATURE BY FEATURE

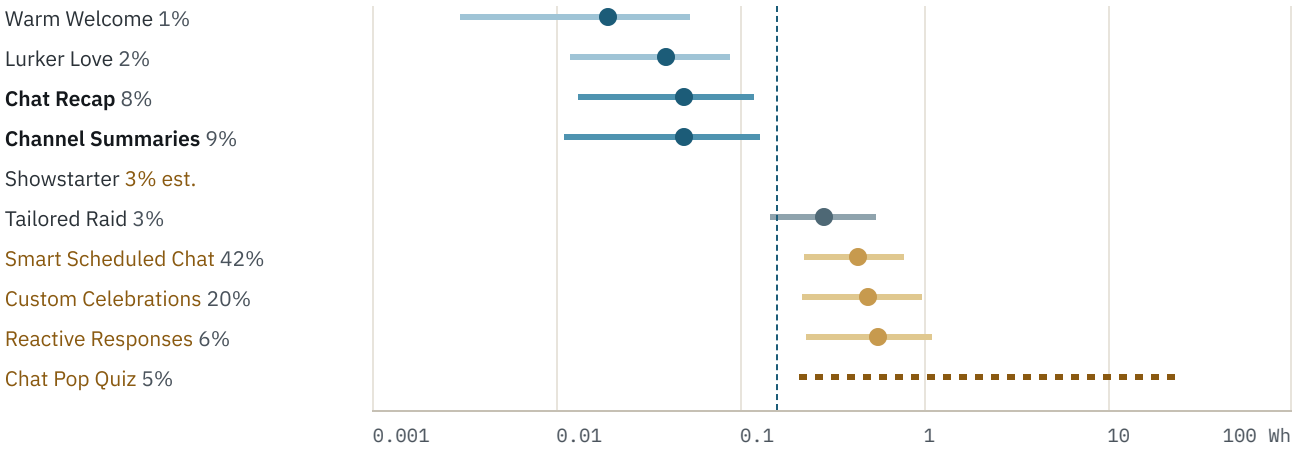
We publish per feature rather than as a single number, because a single number would be wrong. Our shortest routine job and our longest differ by roughly a factor of thirty, and one feature sits far outside even that (§8 names it), so a single average would overstate the common case and understate the rare one. Each row also shows what share of our total AI electricity that feature accounts for, so the figures can be read in proportion to where the energy actually goes.

FEATURE	SHARE OF AI ENERGY	TOKENS IN / OUT	WH	RANGE
Smart Scheduled Chat	**42%**	3,477 / **1,711**	**0.44**	0.23–0.77
Custom Celebrations	**20%**	5,684 / **1,885**	**0.50**	0.22–0.98
Channel Summaries	**9%**	875 / 92	**0.050**	0.011–0.13
Chat Recap	**8%**	803 / 111	**0.049**	0.013–0.12
Reactive Responses	**6%**	6,134 / **1,858**	**0.56**	0.23–1.12
Chat Pop Quiz see §8	**5%**	**111,000** / 1,076	*see range*	**0.2–25**
Showstarter	**3% est.**	not yet logged	*no central value claimed*	not instrumented
Tailored Raid	**3%**	888 / 276	**0.29**	0.15–0.55
Lurker Love	**2%**	568 / 107	**0.039**	0.012–0.088
Warm Welcome smallest	**1%**	379 / 25	**0.019**	0.003–0.053
Average per generation	**100%**	—	**0.16**	**0.07–0.32**

Each percentage is that feature's share of our total AI electricity. We have not tabulated the underlying volumes, but we are not treating them as hidden either: combined with the platform total and the per-feature figures, they can be derived to within a few percent from what is printed here. These are generation counts, not user counts, and they do not convert into either: most of what we run is for people who pay nothing, and we do not publish the mix. A feature's share of our electricity and its share of our usage are different quantities, and they diverge sharply: our two most-used features are among the smallest here. Shares are rounded and do not sum exactly to 100. These figures are also a point-in-time measurement of a system under active change, not a permanent state: the efficiency work named at the end of this paper is in progress, feature behaviour shifts as we tune it, and our total moves with usage in both directions. Energy figures include a measured rate of **1.035 attempts per delivered generation**, so about 3.5% of the electricity above is spent on attempts that did not deliver, whether they were retried or the request was dropped. We count that against ourselves rather than reporting only successful work.

FIGURE 2 · THE SPREAD ACROSS OUR TEN FEATURES

Logarithmic axis. Bars are the full measured range; dots are central values. The dashed line is our volume-weighted average.



Ordered by central energy, with each feature's share of our total AI electricity beside it. The result is lopsided: **one feature accounts for 42% of all the electricity we use**, and three account for more than two thirds of it; Chat Pop Quiz is drawn as a dashed range because no central value is claimed for it (§8). Showstarter has no marker because it is not yet instrumented and no central value is claimed for it. **No single line on this chart can stand in for the product**, which is why we publish the table rather than an average alone.

Total AI inference energy, all features

A per-generation figure with no total behind it is the disclosure pattern that has drawn the sharpest criticism directed at far larger companies: a small per-prompt number means little if the volume is withheld [19]. So here is ours. Across more than a million delivered generations in the window, all ten features combined:

ALL AI INFERENCE, 30 DAYS	IN STREAMS	IN HOUSEHOLDS
~170 kWh	**3.5** broadcasts	**6** days
All ten features, every user, one month	Three-hour streams to 200 viewers, equivalent	Of one average US home's electricity

Roughly **two megawatt-hours a year** at the measured rate. We also drop about **1% of requests** without delivering anything, usually because no backend was available; that electricity is spent for no output and it is included in the totals above rather than netted out.

Most published AI energy figures are estimates applied from the outside. Ours are built on direct instrumentation of our own serving path, and it is worth being precise about which half of the calculation is measured and which is not.

WE MEASURE, PER GENERATION	WE DO NOT MEASURE
Every token in and out, taken from the serving provider's own usage accounting rather than our estimate of it. Two features carry a second count of our own that does not agree with it, and the note below the retry table sets out the difference	The electricity our inference draws. No one serving models through a provider API can meter it directly, and we will not claim otherwise
Which model tier handled the work	The electricity our application and database servers actually draw. Shared-tenancy hosting publishes no per-instance power, so nobody running on it can meter their own slice. §5 models it from published hardware specifications instead, and reports it whole rather than folding it into the per-generation figures
Every attempt, including ones that generated output and then failed	
Whether the request was delivered or dropped	

Instead we apply a published watt-hours-per-token rate from peer-reviewed research to our measured token counts.

The rate comes from work by researchers at Microsoft, published in *Joule*, that models inference energy from token throughput and node power [7], corroborated by an independent academic measurement leaderboard that meters real hardware [14]. We deliberately calibrate against measurements of models *larger* than the compact tier that handles 98% of our work, which makes our figures an upper bound rather than a favourable estimate. And we carry two bounds rather than one, because the research offers two defensible treatments of how input context should be charged. Each of those coefficients also carries its own published uncertainty band. The ranges in §3 run from the decode-only bound at the bottom of its band to the effective-token bound at the top of its, so they cover both the treatment of context and the precision of the measurement behind it.

The two coefficients are these, and we publish them so anyone can reproduce every number in §3 from the token counts beside them:

BOUND	WH PER TOKEN	PUBLISHED BAND	WHAT IT CHARGES FOR
Decode-only lower	0.000200 per output token	0.000114 to 0.000314	Output only. The published model's main analysis, accurate within 6.5% at 500 input tokens [7]
Effective-token upper	0.0000824 per input + output token	0.0000471 to 0.000129	Charges an input token as much as an output token, which overstates the cost of reading context [7]
Frontier tier one feature	0.001033 per output token	0.000533 to 0.002	Applied only to Tailored Raid, the one feature that uses a frontier-class model [7]

Each feature's central figure in §3 is the midpoint of the first two, multiplied by its own measured retry rate. Those retry rates differ sharply by feature, and publishing them is part of the accounting: our least reliable feature needs **1.70 attempts** for every quiz it delivers, while our two highest-volume features need **1.01**.

FEATURE	OUTPUT MEDIAN	OUTPUT MEAN	OUTPUT 90TH PCT.	ATTEMPTS PER DELIVERY
Channel Summaries	91	92	115	1.012
Chat Recap	110	111	130	1.009
Smart Scheduled Chat	1,606	1,711	2,356	1.151
Lurker Love	106	107	128	1.009
Warm Welcome	25	25	31	1.011
Custom Celebrations	1,918	1,885	2,924	1.002
Reactive Responses	1,848	1,858	2,401	1.083
Tailored Raid	272	276	328	1.000
Chat Pop Quiz	905	1,076	2,078	**1.697**

Showstarter records no token counts, one of two open gaps in our instrumentation, and §10 commits to closing it. Its share of our electricity is estimated from its peers, but we claim no per-generation figure for it, on the same basis that we claim none for the quiz: where we cannot measure, we do not publish a central value. Note that we use **means, not medians**, throughout: the published anchors report medians of right-skewed distributions, and multiplying a median by twelve would understate a twelve-generation session. Our own means and medians sit within a percent of each other on the short features and diverge by 19% on the quiz, which is exactly where a skewed distribution would show up.

Two features carry a second, higher token count from our own per-feature tables, differing from the provider's accounting by about twofold. We use the provider's figures throughout. Had we used ours, the platform mean would be about 0.22 Wh rather than 0.16. We name it because it is the largest unresolved discrepancy in our instrumentation, and §10 commits to closing it.

How energy is allocated, and why the two sides differ. The published inference research measures data-centre energy and excludes the network and end-user devices [5][7], whereas the streaming figures in §2 are *dominated* by exactly those things. That is not an oversight. It follows from a choice we make deliberately, and it is worth stating outright.

The boundary has two halves, and they use different conventions on purpose. **At the edge we are incidental**: a chat recap adds essentially nothing at a device that is already on, so we charge no endpoint or network term. Billing it a share of hardware that would be drawing power regardless would double-count energy the broadcast already owns. **In the data centre we are allocated, not marginal.** The published coefficient we apply is an allocated figure by construction: it carries a scaling factor for idle machine capacity and a data-centre overhead multiplier for cooling and power distribution, both of which would be drawing power whether or not any single generation happened. We are marginal at the edge and allocated in the data centre, and we say so because the two conventions do different work. The same reasoning applies to our own servers in the other direction: they carry much more than the AI, which is why §5 reports them whole and separately rather than carving out an invented AI slice.

The stream is charged on a **full** basis: every viewer's screen, every router, the entire delivery path. For a primary activity, that is the correct treatment, because watching is the reason those devices are switched on at all. The same asymmetry applies to the search comparison, and in our favour: the AI-assisted search figure in this paper includes a network and device allowance while our own figures include none. The incidental-endpoint logic above applies to the reader's own screen during a search too, so the gap is smaller than it looks, but it is there, and stripping that allowance out would move the search figure from 1.2–2.1 Wh to roughly 1.1–1.9 Wh.

Using two conventions on two sides of one comparison is a judgement, so the objection is stated here plainly: a critic can argue those televisions would have been on anyway, showing something else, in which case the stream's truly marginal cost is below 49 kWh and every ratio here narrows. We think primary-versus-incidental is the right distinction and we have applied it consistently in both directions. A reader who rejects it now knows exactly which number moves.

5 THE SERVERS BEHIND THE AI

Every FrostyTools figure so far covers AI inference only, the model work itself. That is the right scope for the question people ask, but the inference does not run on nothing: it is requested, queued and delivered by our own servers. So here is that layer too.

FrostyTools runs more infrastructure than this section counts. What follows is the server layer behind the text generation measured in this paper: **two instances**, a 16-vCPU shared-tenancy application server and a dedicated-CPU database server with eight physical cores. Both carry far more than AI: overlays, dashboards, scheduling and the chat integrations run on them too. Treat this as that layer, not as a total for the company.

COMPONENT	POWER	BASIS
Application server 16 vCPU, shared tenancy	58 W	16 of the 96 hardware threads on its host — a sixth of a 48-core-class server processor — against an assumed 350 W of full-node IT draw at moderate load
Database server 8 dedicated physical cores	58 W	The same one-sixth share of an identical host, held exclusively rather than shared
Data-centre overhead cooling, power distribution, lighting	× 1.13	Our provider's published average PUE, which is well below the 1.55 global average reported for 2022 [20] — and using it flatters us, which is why we name it
These two instances	**≈ 130 W**	Continuous, around the clock. **≈ 95 kWh per month**; plausible band 66–139 kWh across the assumptions above

> **The two servers that carry FrostyTools text generation draw about 130 watts between them. One streamer's gaming rig draws about 550 watts.**
>
> Roughly a quarter of the power of a single streaming setup, running continuously. This is the text-generation layer only, not all of FrostyTools.

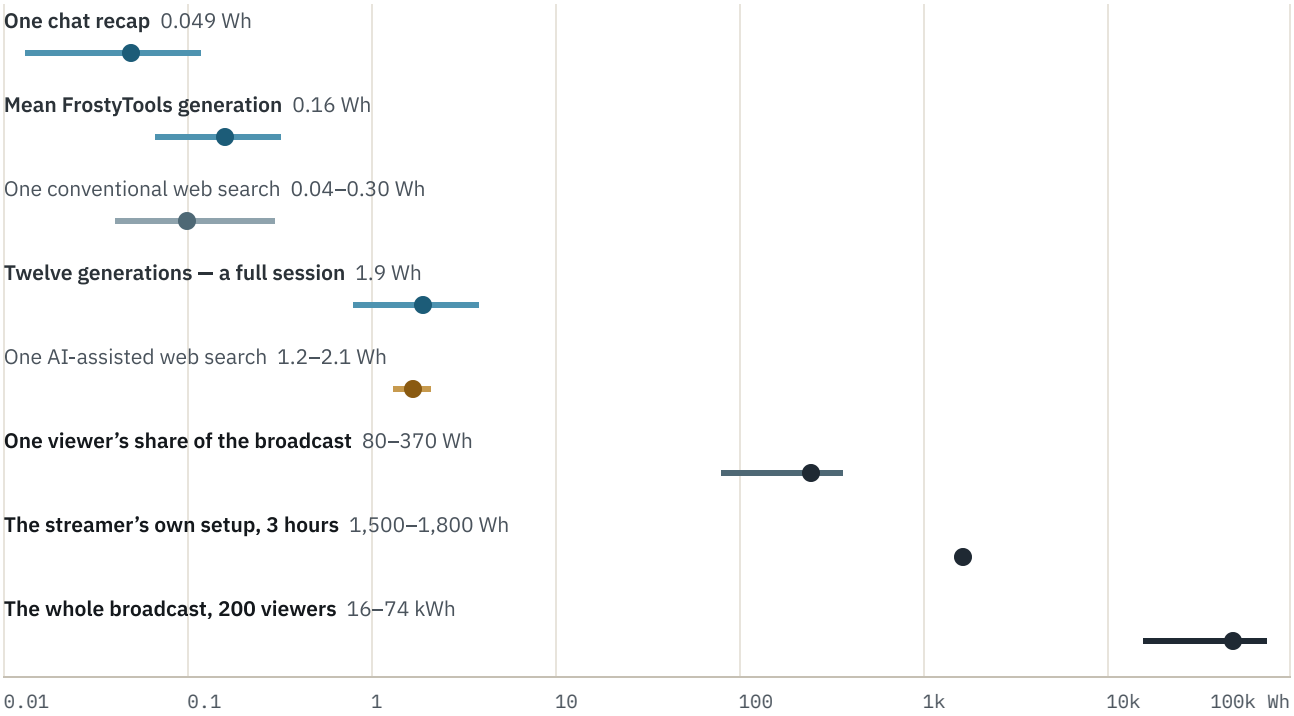
Adding these two servers to the inference gives a combined total of about **265 kWh a month**, or roughly **3.2 MWh a year**, on a plausible band of 235–310 kWh monthly. That is about **five and a half** three-hour broadcasts to 200 viewers, or nine days of one average household's electricity, for the AI work and the two servers that carry it, serving every user on the platform. Other FrostyTools infrastructure is not in this figure.

We do not fold this into the per-generation figures, and the reason matters. Those two servers carry much more than the AI, and we have **no defensible key** for splitting them between AI and everything else they do. Rather than invent one, we report the infrastructure whole and separately. On the most unfavourable reading, charging every watt of these two instances to the AI features alone, a generation costs about **0.25 Wh** rather than 0.16 Wh. That is the honest ceiling, and it changes none of the comparisons in this paper.

> **These wattages are modelled, not measured.** Shared-tenancy hosting publishes no per-instance power draw, so nobody running on it can meter their own slice; we cannot, and we will not pretend to. Four assumptions carry the figure: the host processor's core count and rated power, full-node draw at moderate load, the share of a host our instances occupy, and the provider's PUE. Each is stated above so it can be substituted. The three anchored to published specifications are the first, third and fourth; **the 350 W full-node figure is our own estimate** and is the weakest input here.

FIGURE 3 · FROM ONE RECAP TO ONE BROADCAST

Each gridline is ten times the last. Bars show ranges; dots are central values.



A logarithmic scale is the only way to fit these on one page, and it flatters the small end: the distance from a chat recap to a whole broadcast looks like a few fingers’ width and spans six full decades of the scale. Note that a full twelve-generation FrostyTools session lands inside the range of a single AI-assisted web search, in its upper half.

COMPARISON	DIFFERENCE	NOTE
A streamer's own setup vs. a twelve-generation session, same three hours	**870×**	One person's equipment against one person's tool use. No audience on either side.
One viewer's share vs. a twelve-generation session	**130×** 40× at the low end	One audience member's allocated share, three hours.
One AI-assisted search vs. one chat recap	**24–40×**	Holds for our two most-used features, which are more than half of everything we run.
Recaps needed to equal one viewer's three hours	**5,000**	One every two seconds, for the whole broadcast.
All FrostyTools work in a broadcast, as a share of the broadcast	**0.004%**	Equivalently, the broadcast burns as much in under half a second. Stated as a share rather than a ratio, because a ratio between an audience-wide total and one job invites false precision.

7 WHAT A DAY OF SEARCHING COSTS

Watt-hours are unfamiliar units, and 0.05 Wh means nothing without something ordinary beside it. Searching the web is the most ordinary digital thing there is, and it is worth knowing what it costs front to back, including the work an AI answer does before it ever appears.

A search with an AI answer is not one search. The operator documents that its AI features fan a single prompt out into multiple background searches [6], and the published measurements put that at **9 to 12** on average, with a maximum of 28 observed [18]. With the answer generation on top, one AI-assisted search comes to roughly **1.2–2.1 Wh**, against **0.04–0.30 Wh** for a plain keyword search. The full build-up, its wide band and the two costs nobody counts are set out with the sources at the back of this paper; the AI-assisted search figure is the least certain number here, and we mark it that way wherever it appears.

A per-search figure is still abstract, because almost nobody runs one search. So here is an ordinary day. The number of searches is a stated scenario, not a measurement, and another count can be substituted.

IN ONE DAY	ELECTRICITY	ARITHMETIC
Seventeen ordinary keyword searches	1.7 Wh	17 × 0.10 Wh, the central figure
Three that return an AI answer	5.1 Wh	3 × 1.7 Wh — each firing 9 to 12 searches behind each typed query
Twenty searches, one day	**≈ 7 Wh**	Band 4.3–11.4 Wh across the published ranges. If half the twenty searches returned an AI answer, the day would be about 18 Wh — nine full sessions

Seven watt-hours is **about four full FrostyTools sessions**, or roughly **140 chat recaps**, spent before anyone has watched anything, opened a game or made a coffee. Three searches out of the twenty carry about **three quarters** of that day's total, because of the fan-out behind each AI answer. Set against watching, it stays small: a whole day of searching is roughly a thirty-fifth of one viewer's three hours of a stream.

> **This is context, not a defence.** That searching costs energy is not an argument that ours does not count, and we are not claiming a FrostyTools generation replaces a search or anything else; the next section says so plainly. It is here for one narrow reason: a number this small is unreadable without an everyday thing beside it, and the most everyday digital act there is turns out to cost more per action than the tools people are asked to feel guilty about.

8 WHERE WE WOULD PUSH BACK ON OURSELVES

Five qualifications sit behind the figures above, and we would rather state them than leave them to be found. A comparison that only ever favours the party publishing it is not evidence.

1 Chat Pop Quiz costs far more than anything else we run

To write a quiz it reads the recent chat backlog, which means about **111,000 tokens of input** per generation, more than two hundred times what the published research assumes. Its energy lands between **0.2 and 25 watt-hours**, and we claim no central value, because the research we rely on is calibrated only to 20,000 input tokens and its authors warn that costs rise sharply beyond that [7]. Every total in this paper nonetheless needs some value for it, so here is the one we use: **the midpoint of our two bounds, about 8 Wh**. We do not present that as a central estimate, because the band spans two orders of magnitude and the published method is calibrated to a fifth of this feature's input length. It is under 1% of our volume, so it moves the platform average by about ±6%, but it is comfortably the most expensive thing FrostyTools does and we would rather name it than let it hide inside an average. It is also our clearest efficiency target.

2 Three features generate ten times what a recap does

Smart Scheduled Chat, Custom Celebrations and Reactive Responses each produce roughly 1,700–1,900 output tokens against a recap's 111, and cost about ten times as much. In Reactive Responses, most of that output is reasoning the viewer never sees, and it is generated and paid for regardless. Together these are about a quarter of our volume. The favourable comparison against AI-assisted search does not describe them.

3 Volume is what scales, not unit cost

A sixth of a watt-hour is trivial on its own. It is not trivial when multiplied by a growing user base at a rising rate, and a per-generation figure is precisely the framing that hides that. This is why we publish our total AI inference energy in §3 rather than the per-generation number alone.

4 A small number is not an environmental claim

This paper measures watt-hours during use. It does not convert those watt-hours into carbon, and here is why. Our servers run in Oregon, on a grid dominated by Columbia River hydroelectric power. Our provider reports fully renewable supply at its European sites rather than its US ones, so **we do not claim that for ourselves**. We chose a US region for two reasons: latency for the streamers who use these tools, and cost, which we pass on in what we charge. A European region would carry a documented renewable supply, higher prices and worse performance. That is the trade-off, and it is ours to own. As we add capacity and regions, documented renewable supply is one of the criteria we weigh, alongside latency and what it costs the people who use this, and we will not pretend it has been the deciding one so far. Where a provider or a region can carry the same performance on a cleaner supply, that is where we would rather be.

The larger share of the energy in this paper is not on our hardware at all. AI inference runs on a serving provider's infrastructure in a region we neither choose nor see, so its grid mix is unknown to us, which means the carbon intensity of the smaller half of our footprint can be stated and the bigger one cannot. Publishing a total would mean inventing the dominant term, so we have not. This paper also excludes water, the manufacture of every device involved, and the energy used to train the models. **We also make no claim that using FrostyTools displaces energy anywhere else.** If a generation is additive, meaning work that would not otherwise have happened, then the honest comparison is against zero, not against a stream, and every ratio in this paper stops being the relevant one.

5 Energy is not a cost structure

These figures are an energy accounting, not a cost structure, and the two do not track each other. The most expensive thing we run is free to use. The two features we run most are among the cheapest. Nothing here should be read as what any part of this product costs to provide or what it earns.

9 IN EVERYDAY TERMS

This section gives the same quantities as household uses, on the arithmetic that watt-hours equal watts times hours. Every row shows the wattage it assumes, because appliance draw varies by model and setting. The ladder runs from the smallest thing we do to the largest thing a streaming year costs.

FIGURE 4 · ONE CHART, THREE ZOOM LEVELS

One chart, read top to bottom. Every band is a true linear scale, and each band expands the bar that was too small to read in the band above.

FrostyTools AI An everyday household thing **Streaming**

ZOOM ×1 — FULL WIDTH = ONE BROADCAST, 49,000 WH shortest at the top

A full session — twelve generations **1.9 Wh** 39 recaps

0.004% of the bottom bar — thinner than this page can print

One viewer’s three hours of watching **245 Wh** **≈ 5,000 recaps**

The streamer’s own rig, one stream **1,650 Wh** ≈ 34,000 recaps

One load of laundry, dried **2,500 Wh** ≈ 51,000 recaps

The whole broadcast, 200 viewers **49,000 Wh** **≈ a million recaps**

↓ The second bar above is too small to read. The next band expands it to full width.

ZOOM ×200 — FULL WIDTH = ONE VIEWER’S THREE HOURS, 245 WH

One chat recap **0.049 Wh** **1 recap**

still one five-thousandth of the bottom bar

A full session — twelve generations **1.9 Wh** 39 recaps

Toasting one slice of bread **40 Wh** ≈ 820 recaps

One viewer’s three hours of watching **245 Wh** **≈ 5,000 recaps**

↓ The second bar is still barely visible. One more expansion, and only FrostyTools is left.

ZOOM ×25,800 — FULL WIDTH = ONE FROSTYTOOLS SESSION, 1.9 WH all FrostyTools

A warm welcome message **0.019 Wh** 0.4 recaps

One chat recap **0.049 Wh** **1 recap**

A full session — twelve generations **1.9 Wh** 39 recaps

Every band is a plain linear scale, so bar length means exactly what it appears to mean, and that is why FrostyTools has to be magnified twice before it becomes visible at all. Within each band the shortest bar is at the top and the band's own reference value fills the bottom. The mental model worth taking away: **every individual thing FrostyTools does sits below a slice of toast**, while everything about streaming sits alongside household appliances: one viewer's three hours is half a dozen slices of toast, and the broadcast as a whole is twenty loads of laundry. Recap counts are approximate: one chat recap is 0.049 Wh with a measured range of 0.013–0.119 Wh, so the larger counts here are good to roughly a factor of two either way and should be read as orders of magnitude, not exact tallies.

The table below gives the same quantities with every assumption exposed.

AMOUNT	EQUIVALENT TO	ASSUMPTION
0.019 Wh a warm welcome message	A 60 W bulb for about **1 second**	60 W incandescent bulb [9]
0.049 Wh one chat recap	A games console for **under a second**, a 60 W bulb for about **3 seconds**, or a television for **under 2 seconds**	200 W games console [10]; 60 W bulb [9]; 100 W television [2]
1.9 Wh a full twelve-generation session	A games console for about **35 seconds**, a 60 W bulb for about **2 minutes**, or a microwave for **7 seconds**	200 W games console; 60 W bulb [9]; 1,000 W microwave [10]
40 Wh toasting one slice of bread	About **800 chat recaps**, or **250 generations** of any kind at our average	1,200 W toaster for two minutes [10]
245 Wh one viewer's three hours	A television for **2.5 hours**, or **5,000 chat recaps**	100 W television [2]
1.65 kWh the streamer's own rig, one stream	A space heater running for **about an hour**, or **34,000 chat recaps**	1,500 W space heater [10]
49 kWh one broadcast, all 200 viewers	**1.7 days** of an entire household's electricity; a space heater running **33 hours**; **20 loads of laundry** dried; about **4–9 weeks** of a refrigerator	Household 10,500 kWh/yr; 1,500 W heater [10]; 2–3 kWh per dryer cycle [12]; 300–600 kWh/yr fridge [11]
10,200 kWh a year of streaming, four a week	**An entire US household's annual electricity.** The FrostyTools running alongside all of it: about **0.4 kWh** — one 60 W bulb left on for **7 hours**, for the entire year	208 broadcasts × 49 kWh; twelve generations per broadcast; 60 W bulb [9]

These conversions are for scale, not appliance audits, and should not be quoted without the assumption attached. Colour coding follows Figure 4: unshaded rows are FrostyTools; shaded rows are everyday household uses and the streaming side. The final row is a summary and carries both sides at annual scale.

Five commitments follow from what this measurement exercise exposed, and we would rather list them here than be asked for them. None of them is finished. Measuring this is now part of how the product gets built, which is what makes the list a working one rather than a statement of intent.

Close the missing token counts. One feature, about 2% of our volume, does not yet record token counts. Its share in §3 is estimated from its peers and we claim no per-generation figure for it at all. That is a one-line fix and it should not have been outstanding when this paper was written.

Reconcile the two token sources. Two features carry counts from our own tables that differ from the provider's by about twofold, and we cannot yet say which is right. Until they agree, the platform mean carries an unresolved band of 0.16 to 0.22 Wh. It is the largest open question in our instrumentation and it is the next one we close.

Log every attempt on every feature. Our 3.5% failover overhead is measured on the features that record every attempt and inferred for the rest, which makes it a floor rather than an exact figure. We would rather report a number we can stand behind completely.

Replace the modelled server figures. The infrastructure wattages in §5 rest on a full-node power estimate we cannot verify from inside shared-tenancy hosting. They are the weakest part of this account, and the first thing we would replace with a measurement if per-instance draw became available.

Reduce the quiz. It is comfortably the most energy-intensive feature we run, and the largest single efficiency opportunity in the product. Work on it is in progress.

THE SHORT VERSION

The two things FrostyTools does most, recapping chat and summarising a channel, are together more than half of everything we run, and each uses about as much electricity as a single web search, and a full working session sits inside the range of a single AI-assisted search, in its upper half. The broadcast they run alongside uses more electricity in half a second than a full session of them uses in three hours.

What this paper will not claim is that every FrostyTools feature costs the same. Four of them cost considerably more than a recap, one of them a great deal more, and §8 says which.

ABOUT FROSTYTOOLS

Most Twitch chatbots moderate chat, run timers and answer commands. FrostyTools writes the chat itself. It exists to fight the uncertainty, disconnect and creative block that every streamer runs into, and every feature below is named for the problem a streamer described. The brief on all of them is the same: notice people, do not stand in for the streamer. At a TwitchCon 2024 panel on the ethical use of AI in streaming, FrostyTools was cited as an example of doing it well.

FEATURE	WHAT IT IS FOR
Warm Welcomes	New followers arrive unacknowledged and never come back. Each one gets a personalised greeting, the first impression that turns a curious newcomer into a regular.
Full-Scope Shout-Outs	A bare `!so` says nothing. This reads the channel being pointed at and says something true about it, whether that streamer blind-raided you or you have known them for years.
Lurker Love	Lurkers hold a stream up and are rarely seen. `!lurk` gets a reply referencing what that person actually said earlier.
Custom Celebrations	Sub and bit thank-yous read as copy-paste. These reference what just happened on stream.
Tailored Raid Messages	Raids land as a wall of nothing. This crafts the message on the spot, for you and for your target.
Vibe Raider	Finding somewhere to send your community, live, under time pressure. This surfaces channels matching the vibe and content you are after.
Chat Recaps and Trivia	Ad breaks kill momentum. Recaps answer “what did I miss?” for newcomers and keep chat moving through the break.
Chat Quiz	A Twitch extension. Chat spends bits to challenge you with a gameshow question generated from something said in your own chat. Right or wrong, you earn bits.
Reactive Responses	Dead chat while you are locked into gameplay. Viewers hold real conversations with the bot, in a personality you set.
Showstarter	Going-live anxiety: an empty chat and nothing to say. This opens with a “last time on” recap so early viewers have something to read and you have something to talk about.
Smart Scheduled Messages	Nobody reads the same timer twice. These evolve with the stream.

§3 measures these under their internal names: Warm Welcomes as Warm Welcome, Full-Scope Shout-Outs as Channel Summaries, Tailored Raid Messages as Tailored Raid, Chat Recaps and Trivia as Chat Recap, Chat Quiz as Chat Pop Quiz, and Smart Scheduled Messages as Smart Scheduled Chat. Vibe Raider is a retrieval feature rather than text generation and is not among the ten measured here.

Over 20 personality presets ship with the bot, among them Viking, Pirate and Film Noir Detective. A streamer can also write their own voice, in their own language, with their own channel emotes. Setup is Twitch OAuth and takes under a minute. No tech skills, nothing to download, nothing running on the streaming PC.

The core Twitch chatbot is free. A paid tier unlocks further features and is what pays for the AI measured in this paper. Chat Quiz is free to add.

What this paper does not cover. FrostyTools also makes Highlight Hunter, which turns VODs into memorable clips in minutes rather than hours of scrubbing; Shortify, which cuts them into short-form video; and plug-and-play browser-source overlays for BRB and Starting Soon screens. None of them is measured here. Analysing video is a different and substantially larger kind of work than generating text, and folding it into these figures would misstate both.

What we will and will not claim. The social and ecological weight of this work matters to us, and we build under real trade-offs. We choose server regions for latency and for cost, which we pass on to the streamers who pay for this, and the grid that comes with those regions is not the cleanest one we could pick. The larger share of the energy in this paper runs on infrastructure whose grid mix we cannot see at all. So we will not claim a carbon benefit we cannot show. What we will claim is this: we measured our own AI energy instead of estimating it, published the total rather than only the per-generation figure, and named the feature that costs far more than the rest. Work to reduce it is under way. Documented renewable supply is one of the criteria we weigh as we add regions, alongside latency and cost, and it has not been the deciding one so far. That is what responsibility looks like to us: measuring honestly, fixing what the measurement exposes, and naming the trade-offs we made.

The product, the pricing and the full feature list are at frostytools.com.

SOURCES AND METHOD NOTES

[1] G. Kamiya, International Energy Agency, “The carbon footprint of streaming video: fact-checking the headlines,” 2020. Source of the 0.077 kWh per viewer-hour intensity and the 72 / 23 / 5 split. Drawn from 2019 on-demand viewing; the IEA notes efficiency has improved since. iea.org

[2] ITU-R Report BT.2521-0, “Practical examples of actions to realize energy-aware broadcasting,” 2023. About 18 W for end-to-end delivery; under 1 W for data centres and CDN at 5 Mbit/s HD; under 3 W for a smartphone to about 100 W for a television. itu.int

[3] Google Cloud, “Overview of the Live Stream API” — cited for how live adaptive-bitrate transcoding works. docs.cloud.google.com

[4] U. Hölzle, Google Official Blog, “Powering a Google search,” January 2009. 0.3 Wh per search, stated as including prior work such as building the index. The only first-party per-search figure ever published. googleblog.blogspot.com

[5] Google Cloud, “Measuring the environmental impact of AI inference,” August 2025, and its technical paper. 0.24 Wh for a median assistant text prompt; excludes external and data-centre networking, end-user devices, training and storage. cloud.google.com · arXiv:2508.15734

[6] Google Search Central, “AI Features and Your Website.” Confirms AI answers may issue multiple background searches; publishes no count. developers.google.com

[7] F. Oviedo, F. Kazhamiaka, E. Choukse, A. Kim, A. Luers, M. Nakagawa, R. Bianchini and J. M. Lavista Ferres (Microsoft), “Energy use of AI inference, efficiency pathways, and test-time scaling,” *Joule*, April 2026. The energy model this paper applies, its per-model measurements, and its stated calibration limits. doi.org/10.1016/j.joule.2026.102430 · preprint arXiv:2509.20241

[8] NVIDIA, “GeForce RTX 4070 Ti” product specification, 2023 — the rated-power anchor for the streaming-setup allowance. nvidia.com

[9] US Department of Energy, Federal Energy Management Program, “Purchasing Energy-Efficient Light Bulbs.” Frames the traditional 60 W incandescent bulb by its 800-lumen output, for which it lists 5.9–10.5 W LED replacements. energy.gov

[10] Virginia Cooperative Extension, “Estimating Appliance and Home Electronic Energy Use,” 2020. pubs.ext.vt.edu

[11] ENERGY STAR, Certified Refrigerators product finder. energystar.gov

[12] ENERGY STAR, “Clothes Dryers Key Product Criteria.” energystar.gov

[13] The Shift Project, “Did The Shift Project really overestimate the carbon footprint of online video?” 2020 — disputes the IEA revision in [1], from the opposite direction. Included because the streaming anchor is contested. theshiftproject.org

[14] ML.ENERGY Leaderboard v3.0 — 46 models over 7 tasks, metered on real accelerator hardware. Independent corroboration of the small-model energy band. ml.energy

[15] “Estimating the Increase in Emissions caused by AI-augmented Search,” arXiv:2407.16894. The efficiency-scaled 0.042 Wh conventional-search figure. arXiv:2407.16894

[16] Google 2026 Environmental Report — data-centre electricity above 42 TWh in 2025, used with [17] as a top-down cross-check on per-search energy. blog.google

[17] Google internal data reported January 2025 — more than 5 trillion searches a year. searchengineland.com

[18] Query fan-out measurements. Seer Interactive (501 prompts, Gemini 3 API): 10.7 average, minimum 3, maximum 28. Nectiv (60,000+ captured fan-outs): 9.06 average, maximum 28. Google I/O 2025 material: 12 to 15 average, complex queries past 20. Independent studies converge on 8 to 12. Adopted here: 9 to 12, with an outer band of 5 to 20. Chiefly third-party measurement. The operator has published no per-answer count, and the I/O material is descriptive rather than a disclosure. searchenginejournal.com

[19] C. Crownhart, MIT Technology Review, “Google’s still not giving us the full picture on AI energy use,” 28 August 2025 — the criticism that a per-prompt figure without total volume is not a disclosure. technologyreview.com

[20] Open Compute Project, “Sustainability Metrics Considerations — PUE.” Reports a global average data-centre PUE of 1.55 for 2022, the figure our provider’s published 1.13 is measured against. opencompute.org

ON THE AI-ASSISTED SEARCH FIGURE

No operator publishes a modern per-search energy value, so 1.2–2.1 Wh is our own construction: a conventional retrieval cost of 0.04–0.30 Wh [4][15], multiplied by the 9 to 12 background searches an AI answer issues on average [18], a technique the operator documents without publishing a count [6], plus 0.24–0.70 Wh for generating the answer [5] and a network, router and endpoint term of 0.06 to 0.20 Wh [2]. Taking the widest defensible value for every term, including a fan-out band of 5 to 20 sub-queries rather than the 9 to 12 used above, the full plausible band is 0.5 to 7 Wh.

A top-down check points higher, not lower: 42 TWh of data-centre electricity against 5 trillion searches implies 0.4–1.3 Wh for a conventional search alone at a plausible search share of total compute [16][17]. Two further costs are unquantified by everyone, including us: autocomplete, which issues a backend request on roughly every keystroke, and the crawling and indexing that happen before any search is typed. No figure in this section may be attributed to Google as a disclosure. Google has never published a modern per-search energy value; what appears here is our own reconstruction from published sources, and it is identified as such wherever it is used.

ON THE MAKING OF THIS PAPER

The research and drafting of this white paper were done with the assistance of frontier AI models, under the direction of the author. The work they could not do was the substance of it: interrogating our own production systems, writing and running the queries behind §3, deciding which figures were defensible and which were not, and judging every comparison in here against what we know about how the product actually behaves. Every figure here was validated against the system or the published source it came from. We mention it because a paper about the cost of AI text generation should say when it used some.

REUSE AND CITATION

Yes, you may use this. No permission request needed.

Quote an excerpt	Freely, with attribution.
Republish in full	With attribution and a link to frostytools.com/research/ai-energy-use.
Reproduce a figure or number	Provided the stated boundary, measurement window and uncertainty range travel with it. A figure from this paper without its range is not a figure from this paper.
Translate or adapt	Ask first: frostytools.com/discord.

CITE AS

Rapport, S. (2026). *FrostyTools AI text generation: a measured workload and its operational electricity cost.* Inner Self Labs, LLC. frostytools.com/research/ai-energy-use

All figures are allocated operational-electricity estimates for the period of use, derived from a 30-day measurement window, and exclude equipment manufacturing, data-centre construction and model training. Token counts, model tiers, attempt counts and volumes are measured in production; the energy per token is taken from published third-party research. Per-generation figures are scoped to AI text-generation inference; FrostyTools' video-analysis products are not included in any figure here. §5 adds the two servers that carry that work, modelled from published hardware specifications, and other FrostyTools infrastructure is not counted. §4 sets out how energy is allocated on each side of every comparison, and states the objection to that choice. Volumes and shares are rounded.