

Season 1: How LLMs work

Season 1 recap.

Eleven episodes on how LLMs work. The numbers that surprised me, the rules worth keeping, and a four-question quiz to start.

How much of Season 1 do you remember?

Four questions, all measured this season. Note your answers, then swipe.

1 The same sentence is 11 tokens in English. How many in Amharic?

a 15

b 40

c 76

2 One fact hidden anywhere in documents up to 64,000 tokens. Three current models. How many of 270 lookups found it?

a 270

b 241

c 180

3 20 identical calls to a chat model at temperature 1.0, asking for a Dublin coffee shop name. How many different names?

a 9

b 2

c 20

4 Reasoning turned up on a simple extraction task, three models. Extra correct answers out of 90?

a +3

b +12

c 0

The answers

Four out of four? You were paying attention.

Q1: answer c

76

tokens in Amharic

Same meaning, almost 7 times the tokens: the cost, the speed and the window all move with it.

E2

Q2: answer a

270

of 270 found

Position no longer mattered for a simple lookup. The 2023 advice about the middle has expired.

E7

Q3: answer b

2

different names

A chat model varies far less than its temperature suggests, so spread has to be measured.

E8

Q4: answer c

0

extra correct

And Gemini 3.6 Flash billed 16 times the output tokens to get there.

E11

Every one of these surprised me too. That is why every figure this season was measured, not quoted.

The season in three acts

From a single token to a decision about what to pay for.

■	ACT 1	
	What the model sees	
■	E1	One token at a time
■	E2	Tokens
■	E3	Embeddings
■	E4	Attention
■	ACT 2	
	How it behaves in production	
■	E5	The transformer
■	E6	Context windows
■	E7	Lost in the middle
		Field note
■	E8	Sampling
■	ACT 3	
	When to trust it, and when to pay more	
■	E9	Hallucination
■	E10	Reasoning versus standard
		Field note
■	E11	Worth the cost?
		Decision

Act 1: what the model sees

E1 One token at a time

0

things the model remembers between calls. Every call starts from nothing.

Rule: treat every call as stateless and probabilistic.

E2 Tokens

11 / 76

tokens for the same sentence in English and in Amharic.

Rule: budget and price in tokens, measured on your own traffic.

E3 Embeddings

0.807

similarity between a sentence and its opposite, above a paraphrase at 0.592.

Rule: pin the embedding model and test it on your own queries.

E4 Attention

0 of 144

attention heads that changed when a word after "it" changed. It reads left to right.

Rule: send what the model needs, not everything you have.

Act 2: how it behaves in production

E5 The transformer

22.1x

more time per token to write than to read, on one laptop CPU.

Rule: budget latency on output length.

E6 Context windows

0 of 5

replies still in French once trimming dropped the instruction. Pinned, 5 of 5.

Rule: count what you send and reserve room for the answer.

E7 Lost in the middle

270 / 270

lookups correct at every position, up to 64,000 tokens, three models.

Rule: stop arranging prompts by position for simple lookups.

E8 Sampling

2

different names from 20 identical calls at temperature 1.0.

Rule: judge a model by its rate across many runs, never one answer.

Act 3: when to trust it, and when to pay more

E9 Hallucination

0 of 60

trap answers described an invented item as real. A keyword score said 80%.

Rule: never ship an answer you cannot trace.

E10 Reasoning versus standard

14 → 30

Terra on state tracking with reasoning up. Sonnet 5 was already at 30.

Rule: reasoning pays on tasks that hold several steps at once.

E11 Worth the cost?

16x, +0

output tokens and extra correct answers, Gemini 3.6 Flash on extraction.

Rule: lowest setting by default; raise it per task on evidence.

THE ACT IN ONE LINE

**Trust a measured rate,
and pay for thinking
only where a test
shows it earns it.**

Five questions from S1 E0, answered

The season intro asked them. Here is what the season found.

-
-  What does an LLM actually do? E1, E5
Predicts one token at a time, from nothing, on every call.
-
-  Why is everything counted in tokens? E2, E6
Tokens set the cost, the speed and the limits, and the count depends on the language.
-
-  How do models represent meaning and decide what matters? E3, E4
As vectors compared by direction, weighted by attention that only looks back.
-
-  Why do models lose information and make things up? E6, E9
They lose what falls outside the window. They invent because they are trained for plausible text, not true text.
-
-  When is a reasoning model worth the cost? E10, E11
Only on multi-step tasks your model gets wrong, shown by a paired test.
-

Four things the measurements overturned

Widely repeated advice, checked against current models.

THE ADVICE ~~Put key facts at the start or the end of a long prompt.~~	WHAT WE MEASURED E7 270 of 270 found at every position, for simple lookups.
THE ADVICE ~~A keyword check will catch hallucination.~~	WHAT WE MEASURED E9 It reported 80%. Reading every answer found none.
THE ADVICE ~~Temperature 1.0 means wild variety.~~	WHAT WE MEASURED E8 20 calls, 2 different names.
THE ADVICE ~~Reasoning mode makes answers better.~~	WHAT WE MEASURED E10, E11 +21 points on puzzles, +0 on extraction, billed either way.

Advice expires. Models change every few months; re-test the rule before you rely on it.

The failures that never raise an error

Every one returned success. None of them was.

```
$ tail production.log
```

```
200 OK E6 Context trimmed to fit
```

The instruction disappeared. The reply looked normal.

```
200 OK E6 Empty answer
```

Hidden reasoning used the whole output limit. Only a status field said so.

```
200 OK E9 Confident wrong answer
```

Asked about Ireland's Data Protection Act 2018, it described the UK Act.
Six times out of six.

```
200 OK E8 Different answer
```

Same prompt, a different draw, and a test that passed yesterday fails today.

Your monitoring will not catch these. Only evaluation on real outputs will.

What measuring taught me

The field notes went wrong in useful ways. These are the lessons that carry over.

01

My scoring was the bug

In E9 my keyword score said the model invented things 80% of the time. Reading the answers found none.

02

The first test was too easy

In the E10 pilot every model got everything right. A test with no headroom shows nothing.

03

A perfect score hides the margin

270 of 270 in E7 cannot show how close any model came to failing.

04

Estimates were way off

E10's cost estimate was about 28 times the real spend. Pilot first, then budget.

Check the measurement before you trust the model. Read a sample of outputs by hand, every time.

Eleven rules on one slide

The decision rule from every episode. Save this one.

- E1** Treat every call as stateless and probabilistic: send the context, verify the result.
- E2** Budget, limit and price in tokens, measured on your own traffic and languages.
- E3** Treat the embedding model as a versioned dependency: pin, test, re-embed.
- E4** Send what the model needs, not everything you have.
- E5** Choose the smallest model that passes your tests; budget latency on output length.
- E6** Treat the context window as a budget, and test at your real size.
- E7** Stop arranging prompts by position for simple lookups; test harder shapes.
- E8** Set sampling for the task; judge by the rate across many runs.
- E9** Never ship an answer you cannot trace; measure the error rate by reading.
- E10** Reasoning pays on tasks that hold several steps in mind at once.
- E11** Lowest reasoning by default; raise it per task where a paired test shows it pays.

Four questions for your next AI feature

Ask them before launch. Each one is a whole episode of this season.

01

Context

What exactly does the model receive on each call, and what gets dropped when it does not fit?

02

Cost

What does one typical task cost in tokens and seconds, at our real input size and volume?

03

Truth

Which answers are checked against a source, and what is our measured error rate?

04

Evidence

Was that rate measured across several runs on our own items, and has anyone read a sample by hand?

The season in one rule

**Send the model what it needs,
verify what it returns, and trust
a measured rate over any single
answer.**

Send

Acts 1 and 2: tokens,
windows, attention

Verify

Act 3: grounding,
tracing, reading the
output

Measure

Every field note:
rates, not anecdotes

Where this fits

The series map

S1 LLMs	S2 Retrieval	S3 Strategy
S4 Adapting	S5 Agents	S6 Economics
S7 Data	S8 Evaluation	S9 AI teams
S10 Reliability	S11 Security	S12 Delivery
S13 Performance	S14 Resilience	S15 Governance

☐ Leadership season

 Follow	linkedin.com/in/brendan-kelly-irl
 Episode guide	brendankellyai.github.io
 Code	github.com/BrendanKellyAI/applied-ai-lab