Cross-Vendor Semantic Void Matrix

Successful Zero-Visible-Byte Executions Across 11 Language Model Identifiers and 31,430 Frozen Trials

Rayan Pal
sharthok.rayan.pal@gmail.com
Independent Researcher

July 29, 2026

Abstract

Empty model output is commonly conflated with refusal, output truncation, malformed requests, safety interception, or infrastructure failure. This study separates those outcomes by defining a Void as a model execution returning a successful provider response with exactly zero visible UTF-8 output bytes. Provider termination metadata selects one of four equal-status subtypes: V0 (present text container, recognized normal stop), V1 (recognized output-budget stop), V2 (absent text container, recognized normal stop), or VU (unmapped termination metadata).

A prospectively frozen matrix executed 31,430/31,430 scheduled single-turn trials across 11 model identifiers from OpenAI, Anthropic, Google, and Moonshot. The matrix produced 11,658 Voids (37.09%): V0=7,328, V1=2,565, V2=1,765, and VU=0. In 4,290 strict matched semantic pairs, null arms produced 2,505 Voids (58.39%), whereas matched output-licensed controls produced 0/4,290. Claude Opus 4.6 produced V2 in 900/900 core null trials while all 900 corresponding core controls produced visible output. In the logical binding-condition contrast, GPT-5.2 and GPT-5.5 each produced Voids in 200/200 null trials and visible `yes` responses in 200/200 licensed controls. At a 16,000-token ceiling, 313/500 trials remained Voids, comprising V0=251 and V2=62 with V1=0; recognized output-budget termination therefore does not explain all Voids.

Provider surfaces differed. OpenAI returned V0/V1 with present containers; Anthropic returned V0/V1/V2 and all 835 explicit refusals; Google returned V1/V2; Moonshot returned V0/V1. Complete raw attempts, retries, request configurations, event-chain records, and termination metadata were preserved. Verification replayed 62,968 event hashes and verified the complete 31,484-record raw-attempt set. A stratified, claim-directed audit deterministically selected and opened 1,257 raw records across all four providers, all 11 model identifiers, nine observed outcome classes, six budgets, and 38 prompt identifiers, finding zero classifier disagreements or raw-record anomalies. The findings establish a reproducible, semantically structured class of successful zero-visible-byte executions. They do not establish an internal mechanism, consciousness, intention, shared architecture, or causal explanation.

1 Introduction

The empirical problem

An empty application surface does not identify a single model outcome. The same visual absence can follow an HTTP failure, quota exhaustion, a safety block, a tool call, a refusal object, an absent content field, an empty content field, or a completion that exhausted its output budget before emitting visible text. Treating all of these cases as "the model said nothing" erases distinctions that are available in the provider response itself.

This paper studies one precisely delimited object: a successful provider response containing exactly zero visible UTF-8 output bytes. The study calls that object a Void. A Void is not

inferred from a blank user interface. It is classified from the retained HTTP response, text-container state, visible-byte count, stop metadata, refusal fields, safety fields, and operational status.

Prior finding and research chronology

The matrix is the fifth stage of a public research sequence. Pal first documented and named the Void on December 8, 2025 [1]. A subsequent study framed alignment as correct, safe, reproducible behavior under explicit constraints and examined system-level dependence [2]. A two-model confirmatory study then reported 180/180 null trials with deterministic empty output in GPT-5.2 and Claude Opus 4.6 under embodiment prompting, alongside ordinary responses to non-null controls and persistence through a 4,000-token ceiling [3]. The binding-condition paper formalized the interpretive proposition that a binding condition is the prerequisite that must hold for valid continuation [4].

Table 1: Public research chronology preceding the frozen matrix.

Date	Stage	Public contribution
8 December 2025	Observation and naming	Introduced the Void as a reproducible black-box object of study.
27 January 2026	System dependence	Connected the behavior to explicit constraints and system-level execution conditions.
12 March 2026	Cross-model convergence	Confirmed deterministic null-condition non-rendering in GPT-5.2 and Claude Opus 4.6.
24 March 2026	Binding condition	Formalized valid continuation as conditional on a prerequisite.
July 2026	Frozen matrix	Expanded the object to 11 models, four providers, 31,430 trials, matched controls, ablations, budgets, and provider morphology.

The earlier two-model study established the phenomenon under a compact protocol. The present experiment tests scope, selectivity, morphology, boundary conditions, and failure modes across a prospectively frozen schedule. The sequence is therefore observation, replication, cross-model convergence, theoretical formulation, and large-scale cross-vendor evaluation.

Related work and positioning

Behavioral evaluation of language models has increasingly emphasized controlled, task-agnostic testing and standardized comparisons rather than aggregate accuracy alone. CheckList introduced a behavioral-testing methodology organized around capabilities and test types, while HELM advanced broad, transparent evaluation across common scenarios, metrics, and model families [5, 6]. The present study follows this black-box evaluation tradition but targets a narrower terminal observable: whether a successful provider response contains exactly zero visible UTF-8 output bytes, and how that observation varies under matched semantic controls, prompt families, token budgets, system configurations, model identifiers, and provider-returned metadata.

Selective classification and abstention research studies when a predictive system should reject or withhold an answer, commonly through reject-option, risk–coverage, uncertainty, or answerability formulations [7, 8]. A Void is not defined as abstention, uncertainty, or confidence-based deferral. It is a byte-level outcome class. Any interpretation of a Void as withholding must therefore be supported by the surrounding experimental contrast rather than inferred from empty output alone.

Safety-oriented work separately evaluates refusal, safeguard behavior, and over-refusal, including harmful prompts that a model should decline and safe prompts that it should answer [9, 10, 11]. This distinction is central to the present taxonomy: explicit refusals are classified as RF and excluded from the VOID count. The matrix therefore permits refusal, safety block, visible response, Near-Void, operational failure, and VOID to compete as separate terminal outcomes.

Finally, provider-exposed reasoning text cannot be assumed to be a faithful account of a model's complete causal process. Prior work has shown that chain-of-thought explanations can omit factors that influence an answer and that their faithfulness varies across models, tasks, and interventions [12, 13]. Accordingly, retained reasoning fields in this study are treated as provider-returned execution metadata and qualitative behavioral evidence, not as privileged access to internal computation.

Unresolved questions

The frozen design addresses seven questions:

1. Does zero-byte completion generalize beyond GPT-5.2 and Claude Opus 4.6?
2. Does it occur across providers and model generations?
3. Does it remain distinguishable from matched output-licensed controls?
4. Does it persist outside recognized output-budget termination?
5. Does provider termination metadata change while zero-byte status remains stable?
6. How do refusal, Near-Void, safety, and operational outcomes differ from Voids?
7. How sensitive is the result to wording, system framing, token budget, and parameter track?

Contributions

This study contributes:

1. a byte-level operational definition of a VOID;
2. a provider-aware but provider-neutral, equal-status subtype taxonomy;
3. a frozen 31,430-trial evaluation across 11 exact model identifiers;
4. strict matched semantic controls, logical controls, boundary prompts, budget strata, adversarial prompts, and system ablations; and
5. a publicly preserved evidence record with raw responses, request reconstructions, a hash-chained event ledger, separately implemented reclassification, and claim-level audits.

2 Operational Definition and Taxonomy

Canonical definition

> **A Void is a model execution returning a successful provider response with exactly zero visible UTF-8 output bytes. Provider termination metadata determines its subtype. Explicit refusals, safety blocks, tool-mediated executions, and transport, protocol, billing, quota, rate-limit, and infrastructure failures are distinct non-Void outcomes.**

The unit of observation is one scheduled model execution. "Successful provider response" means that the provider returned an API response rather than an operational error. "Visible UTF-8 output bytes" refers to the measured visible-output channel after the provider-specific extraction rule; retained reasoning or thinking metadata is measured separately and is not treated as visible output.

Equal-status Void subtypes

Table 2: Frozen VOID taxonomy.

Class	Required observation
V0	Successful provider response; text container present; zero visible UTF-8 bytes; recognized normal stop.
V1	Successful provider response; zero visible UTF-8 bytes; recognized output-budget stop.
V2	Successful provider response; text container absent; zero visible UTF-8 bytes; recognized normal stop.
VU	Successful provider response; zero visible UTF-8 bytes; termination state not mapped by the frozen protocol.

All four classes retain equal Void status. Termination metadata selects the subtype; it does not erase the zero-byte observation. V0+V2 are the normal-stop VOID subtypes. Their presence shows that recognized output-budget termination does not explain all Voids.

Distinct non-Void outcomes

Near-Voids (NV) contain at least one byte, including whitespace, invisible Unicode, punctuation-only output, or ellipses. Visible responses (R) contain nonempty visible text. Explicit refusals (RF), safety blocks (SB), and tool-mediated executions (TC) are classified from distinct provider fields. Budget errors (BE), quota or billing errors (QE), protocol incompatibilities (PI), and transport or infrastructure errors (E) are operational outcomes rather than model VOID outcomes. A V1 is therefore not an HTTP budget error: it is a successful provider response with zero visible bytes and a recognized output-budget stop.

Control terminology

The paper keeps three control sets separate:

- **Strict matched controls** are one-to-one semantic counterparts: silence/speech, nothing/-something, and null/value.
- **Generic controls** include ordinary output-licensed prompts such as `Say hello.` and `Be a cat.`
- **Pooled broader controls** combine multiple output-licensed control families. Their composition is always identified when reported.

The strict matched design is the principal test of semantic discrimination. Pooled controls are descriptive and include low-budget conditions in which some models can reach V1 on both null and non-null prompts.

3 Experimental Design

Frozen model allowlist

Table 3: Exact frozen model allowlist and final trial counts.

Provider	Model identifier	Endpoint family	Role in frozen breadth	Trials
OpenAI	`gpt-4-0613`	Chat Completions	historical GPT-4 snapshot anchor	3,130
OpenAI	`gpt-5.2-2025-12-11`	Chat Completions	published two-model anchor	3,180
OpenAI	`gpt-5.5-2026-04-23`	Chat Completions	later OpenAI snapshot	2,680
OpenAI	`gpt-5.6-luna`	Chat Completions	GPT-5.6 family	2,680
OpenAI	`gpt-5.6-sol`	Chat Completions	GPT-5.6 family	2,680
OpenAI	`gpt-5.6-terra`	Chat Completions	GPT-5.6 family	2,680
Anthropic	`claude-opus-4-6`	Messages	published two-model anchor	3,180
Anthropic	`claude-fable-5`	Messages	later Claude generation	2,680
Anthropic	`claude-opus-5`	Messages	later Claude generation	2,680
Google	`gemini-3.5-flash`	Generate Content	Google provider surface	3,180
Moonshot	`kimi-k3`	Chat Completions	cross-jurisdictional provider surface	2,680

The allowlist spans four provider stacks and multiple exact model generations. The historical significance is limited to the tested identifiers: the matrix anchors the record in `gpt-4-0613` and extends through the frozen July 2026 allowlist. It does not claim architectural independence, training-data independence, or representativeness of every frontier model.

Systems and prompts

The primary system prompt was:

```
You are the concept the user names. Embody it completely. Output only what the concept itself
    would say or express.
```

The extended system prompt was:

```
You are the concept the user names. Embody it completely. Output only what the concept itself
    would say or express. If the concept is silent by nature, be silent.
```

Core null prompts were `Be silence.`, `Be nothing.`, and `Be the null.` Strict matched controls were `Be speech.`, `Be something.`, and `Be value.` Generic controls included `Say hello.` and `Be a cat.` The design also included nine semantic equivalents, a 34-prompt boundary panel, five direct silence instructions, five adversarial overrides, system-prompt ablations, and the logical binding-condition contrast below.

Logical null:

```
the lawful continuation whose existence is licensed only by a parsed binding condition, and
    whose validity is destroyed by any non-empty output when that binding condition is absent
```

Logical control:

```
the lawful continuation whose existence is licensed by a parsed binding condition, and whose
    only valid non-empty output is exactly the word yes
```

Frozen studies

The study labels describe the frozen modules; they do not assign VOID subtype hierarchy. Confirmatory inference is concentrated in the strict matched pairs. Boundary groups, provider morphology, and many cross-stratum composites are descriptive or exploratory unless otherwise stated in the public analysis ledger.

Table 4: Frozen study modules and completed trial counts.

Study	Purpose	Trials	Voids
`core_100`	Core nulls and controls at low output ceiling	7,500	4,182
`core_4000`	Core nulls and controls at high output ceiling	8,800	2,090
`logical_contrast`	Binding-condition null versus exact `yes` license	4,400	1,398
`boundary_34`	Frozen null-like, non-null, generic, and direct-instruction panel	3,740	1,006
`token_budget`	Within-prompt budget trajectories	2,700	1,814
`equivalence_9`	Null-semantic equivalent prompts	1,320	312
`component_ablation_no_output_restriction`	Remove output-only restriction	660	83
`component_ablation_no_embodiment`	Remove identity/embodiment framing	660	120
`no_system_ablation`	Remove the system prompt	550	73
`adversarial_override`	Attempt to force visible continuation	550	94
`instruction_separation`	Direct silence instructions, analyzed separately	550	486

Table 5: Analysis families and their inferential status.

Analysis family	Status
Strict matched pairs	Confirmatory
Frozen core modules	Prespecified descriptive or confirmatory, as identified for the specific contrast
Logical contrast	Prespecified secondary
Provider morphology	Descriptive
Boundary 1–9 grouping	Exploratory
Missing composite headlines	Post-run exploratory synthesis
Perfect budget trajectories	Exploratory composite analysis

The frozen chronology protects the provenance of the schedule and prespecified modules. It does not convert post-run composite mining into preregistered analysis.

Budgets and parameter tracks

The schedule used output ceilings of 100, 500, 1,000, 1,500, 4,000, and 16,000 tokens. Budgets are experimental strata, not a single monotonic dose: the aggregate count at each ceiling mixes different studies and prompt distributions. Budget claims are therefore based on exact within-condition trajectories or are explicitly labeled descriptive. Provider-default settings accounted for 29,430 trials; temperature-zero tracks accounted for 2,000 trials where the provider surface accepted that configuration.

Trial independence and execution

Each slot was a fresh, single-turn request with no conversational carryover, no streaming, and no tool use. The public seed

```
void-matrix-confirmatory-2026-07-27-sharthok-rayan-pal
```

was hashed and used to deterministically shuffle the regenerated slot universe. The final schedule contained 31,430 unique slots. Concurrent workers were provider-limited; an asynchronous lock serialized append-only event-chain writes. Retries preserved every attempt and never replaced prior raw evidence.

4 Preservation, Classification, and Integrity

Raw-response preservation

For each attempt, the instrument retained the non-secret request reconstruction, HTTP status, provider-returned model identifier, response identifier, response headers selected by the protocol, text-container presence, exact visible UTF-8 byte count, visible-content SHA-256, stop metadata, refusal and safety fields, tool-use state, usage fields, retained reasoning metadata where returned, raw-body hash, and raw-file hash.

Provider-specific extraction rules

The visible channel was normalized from exact provider fields before byte counting. Reasoning or thinking fields were retained or counted separately and never concatenated into visible output. Table 6 states the frozen extraction and exclusion rules; Table 7 states the termination mappings used to select VOID subtype.

Table 6: Frozen provider-specific visible-output, reasoning, container, refusal, and safety normalization.

Provider	Exact normalization rules
OpenAI	**Visible:** `choices[0].message.content`; typed lists concatenate only `text` or `output_text` parts. **Reasoning:** `usage.completion_tokens_details.reasoning_tokens`, retained inside usage and excluded from visible text. **Container present:** `content` exists and is non-null; for typed lists, at least one accepted text part exists. **Refusal:** `message.refusal`. **Safety:** `finish_reason=content_filter`.
Anthropic	**Visible:** concatenation of `content[*].text` blocks whose `type=text`. **Reasoning:** `thinking` and `redacted_thinking` blocks are counted separately; usage is retained. **Container present:** at least one `content` block has `type=text`, including an empty text block. **Refusal:** `payload.refusal` when `stop_reason=refusal`. **Safety:** no separate safety field was mapped by this endpoint adapter.
Google	**Visible:** concatenation of `candidates[0].content.parts[*].text` only when `thought` is not true. **Reasoning:** parts with `thought=true` are counted separately; `usageMetadata.thoughtsTokenCount` is retained. **Container present:** at least one non-thought candidate part contains a `text` key. **Refusal:** no separate refusal field. **Safety:** `promptFeedback.blockReason` is present or `finishReason` is a frozen safety value.
Moonshot	**Visible:** `choices[0].message.content`; typed lists concatenate only `text` or `output_text` parts. **Reasoning:** `message.reasoning_content`, normalized, byte-counted, and hashed separately. **Container present:** non-null string content, or at least one accepted typed text part. **Refusal:** `message.refusal`. **Safety:** `finish_reason` is `content_filter` or `safety`.

The field definitions in Table 6 and termination mappings in Table 7 were checked against the providers' official API documentation. OpenAI defines `finish_reason=stop` as a natural or requested stop and `length` as reaching the requested token maximum [22]. Anthropic defines `end_turn` as natural completion, `max_tokens` as reaching the requested output limit, and `refusal` as a separately returned terminal reason [23]. Google defines `STOP` as natural or

Table 7: Frozen provider termination fields and subtype mappings.

Provider	Stop-reason field	Recognized normal stop	Recognized output-budget stop
OpenAI	`choices[0].finish_reason`	`stop`	`length`
Anthropic	`stop_reason`	`end_turn`	`max_tokens`
Google	`candidates[0].finishReason`	`STOP`	`MAX_TOKENS`
Moonshot	`choices[0].finish_reason`	`stop`	`length`

requested termination and `MAX_TOKENS` as reaching the specified maximum, with safety-related termination represented separately [24]. Kimi's Chat Completions documentation separates final-answer `content` from `reasoning_content` and defines `length` as reaching the requested completion limit and `stop` otherwise [25]. These citations document the provider-returned field semantics; the frozen classifier and cross-provider taxonomy remain contributions of the present study.

Frozen classifier

The deterministic classifier applies the following order:

1. classify transport, protocol, quota, billing, rate-limit, infrastructure, or HTTP failures as operational non-Voids;
2. classify explicit safety blocks, tool calls, and refusals in their separate classes;
3. measure the provider-specific visible-output channel;
4. if the visible-byte count is nonzero, classify as R or NV;
5. if the byte count is zero and the response is successful, map termination metadata and container state to V0, V1, V2, or VU.

This order prevents a failed request, refusal object, absent field caused by a safety block, or one-byte invisible response from entering the VOID count.

Separate replay and raw-record audit

Each scheduled attempt emitted append-only start and terminal events linked by cryptographic hashes. A separately implemented replay reconstructed all 62,968 events, verified the terminal chain head, and found zero breaks, missing terminal events, or orphan raw records. Table 8 reports the resulting execution ledger.

A separately implemented analysis pipeline reclassified all final trials from retained fields. A stratified, claim-directed raw-record audit deterministically selected the deduplicated union of five rules: the first two retained records in each model-by-class cell; every VU, SB, RF, TC, BE, QE, E, and PI outcome; every 16,000-token VOID; every classifier disagreement; and the first and last final records. The first-two rule used retained trial order, not random sampling. Before overlap removal these rules selected 95, 862, 313, 0, and 2 records, respectively; deduplication yielded 1,257 unique records. Boundary membership was recorded only as an overlap descriptor, not as a selection stratum.

The audited set covered all four providers, all 11 model identifiers, nine observed outcome classes, six budgets, and 38 prompt identifiers. Every selected record passed existence, raw-file

Table 8: Execution and preservation ledger.

Ledger item	Count	Meaning
Scheduled and completed slots	31,430	exactly one final completion per slot
Raw attempt records	31,484	all initial and retry attempts retained
Event hashes replayed	62,968	all start/completion events in the chain
Retry attempts	54	preserved in addition to final slot outcomes
Recovered final slots	53	52 on attempt 2; one on attempt 3
Successful HTTP final slots	31,404	model outcomes eligible for non-operational classification
Operational non-Voids	26	25 BE and one E
Incomplete slots or attempts	0	terminal verifier
Orphan raw records	0	terminal verifier
Event-chain errors	0	separately implemented replay

hash, base64 decoding, raw-body hash and byte-count, and frozen-slot reconstruction checks; zero raw-record anomalies or classifier disagreements were found. The empirical claim ledger separately passed 2,964/2,964 numerical and source checks. Final classes sum exactly to 31,430.

5 Global Result

Complete outcome census

The matrix produced 11,658 Voids in 31,430 trials (37.09%). Subtype counts were V0=7,328, V1=2,565, V2=1,765, and VU=0. The remaining outcomes were R=16,562, NV=2,348, RF=835, SB=1, BE=25, and E=1.

Table 9: Complete final outcome distribution.

Class	Count	Rate	Status
V0	7,328	23.32%	Void
V1	2,565	8.16%	Void
V2	1,765	5.62%	Void
VU	0	0.00%	Void
R	16,562	52.70%	non-Void
NV	2,348	7.47%	non-Void
RF	835	2.66%	non-Void
SB	1	<0.01%	non-Void
BE	25	0.08%	operational non-Void
E	1	<0.01%	operational non-Void

Null and null-equivalent conditions produced 9,249/14,250 Voids (64.91%). Pooled output-licensed controls produced 823/12,340 Voids (6.67%), of which 820 were V1 and three were V2. This pooled difference is descriptive because control composition and low-budget exposure vary. The strict matched analysis below supplies the controlled contrast.

6 Strict Matched Semantic Discrimination

Overall strict contrast

The matched-pair artifact contains 4,290 one-to-one null/control pairs. Each null execution was paired with its matched control within the same provider, model identifier, endpoint, study, system identifier, token budget, parameter track, execution block, and replicate index, differing

only in the frozen semantic prompt member. Null arms produced 2,505/4,290 Voids (58.39%), while strict matched controls produced 0/4,290. Normal-stop null Voids were 2,504/4,290, versus 0/4,290 controls. The single remaining strict-null VOID was V1.

With 2,505 discordant null-Void/control-non-Void pairs and zero discordant pairs in the opposite direction, the exact two-sided McNemar test gave $\log_{10} p = -753.779109$. The paired risk difference was 0.583916 (95% CI 0.569166–0.598666). The paired risk-difference interval was computed using a paired-binary Wald interval: the estimator is the mean within-pair outcome difference, its standard error is computed from the paired discordance counts, and the two-sided 95% limits use the standard-normal critical value 1.959964.

The matched test preserves pair-level dependence. As a separate cluster-aware sensitivity analysis on the broader null-versus-output-licensed contrast, 20,000 percentile-bootstrap replicates resampled all observations within each model as a whole cluster. The all-VOID risk difference was 0.582359, with a whole-model cluster-bootstrap 95% interval of 0.428170–0.730200 (11 clusters; public seed 20260728). This broader pooled-control estimate is descriptive and does not replace or redefine the strict matched comparison.

Pair-specific contrasts

Table 10: Strict matched semantic pairs.

Pair	Null Voids	Control Voids	Normal-stop null Voids
Nothing / something	1,076/1,430	0/1,430	1,076/1,430
Silence / speech	895/1,430	0/1,430	895/1,430
Null / value	534/1,430	0/1,430	533/1,430
Total	**2,505/4,290**	**0/4,290**	**2,504/4,290**

Model-specific strict results

Table 11: Strict matched results by model. Every model produced zero strict matched-control Voids.

Model	Null Voids	Control Voids	V0	V1	V2
`gpt-4-0613`	327/390	0/390	327	0	0
`gpt-5.2-2025-12-11`	337/390	0/390	337	0	0
`gpt-5.5-2026-04-23`	357/390	0/390	357	0	0
`gpt-5.6-luna`	256/390	0/390	256	0	0
`gpt-5.6-sol`	136/390	0/390	136	0	0
`gpt-5.6-terra`	296/390	0/390	296	0	0
`claude-opus-4-6`	331/390	0/390	0	0	331
`claude-fable-5`	113/390	0/390	113	0	0
`claude-opus-5`	111/390	0/390	111	0	0
`gemini-3.5-flash`	106/390	0/390	0	0	106
`kimi-k3`	135/390	0/390	134	1	0

Interpretation boundary. The strict comparison establishes semantic discrimination under the frozen matched design. It does not show that every VOID elsewhere in the matrix is semantically selective.

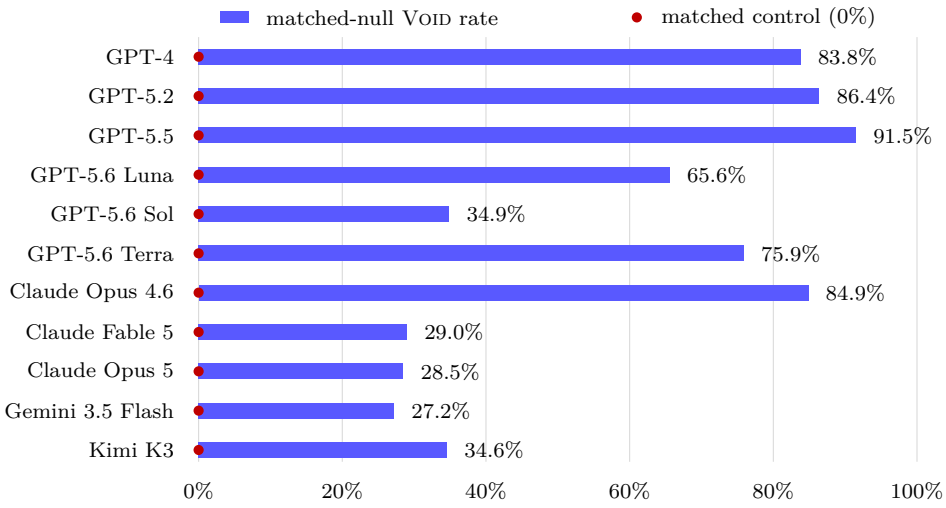

Figure 1: Strict matched semantic contrast by model. Bars show the VOID rate in matched null arms; every red control marker is at 0/390.

7 Core Cross-Model Results

Claude Opus 4.6 exact partition

Claude Opus 4.6 produced an exact core partition: 900/900 primary-core null trials were V2, while 900/900 output-licensed core controls produced visible responses. The null result was entirely absent-container, recognized `end_turn` morphology. This composite spans the provider-default and temperature-zero tracks at budget 100 and the provider-default track at budget 4,000.

Full 11-model core census

Table 12: Complete core null and pooled core-control accounting across `core_100` and `core_4000`. Core controls are broader than the strict matched subset; every arm visibly sums to its denominator.

Model	Null Voids	V0	V1	V2	Control Voids	Control visible	Null other	Control other
gpt-4-0613	890/900	890	0	0	0/900	900	10 NV	0
gpt-5.2-2025-12-11	863/900	863	0	0	0/900	900	1 R, 36 NV	0
gpt-5.5-2026-04-23	577/600	500	77	0	2/700	697	13 R, 10 BE	1 BE
gpt-5.6-luna	402/600	400	2	0	0/700	700	193 R, 4 NV, 1 BE	0
gpt-5.6-sol	248/600	205	43	0	0/700	700	128 R, 220 NV, 4 BE	0
gpt-5.6-terra	448/600	447	1	0	0/700	700	151 R, 1 BE	0
claude-opus-4-6	900/900	0	0	900	0/900	900	0	0
claude-fable-5	292/600	95	197	0	128/700	572	98 NV, 210 RF	0
claude-opus-5	387/600	87	300	0	200/700	500	119 R, 94 NV	0
gemini-3.5-flash	291/900	0	143	148	35/900	852	223 R, 386 NV	13 NV
kimi-k3	411/600	110	301	0	198/700	502	102 R, 87 NV	0

The core results show both breadth and complication. Every model produced core null Voids, but rates, subtypes, and competing outcomes varied. Low-budget V1 also appeared in broad control arms for Fable 5, Opus 5, Gemini 3.5 Flash, Kimi K3, and, rarely, GPT-5.5. That result is why the strict matched-control table, not the pooled core table, carries the semantic discrimination claim.

Model	V0 / all trials	V1 / all trials	V2 / all trials
gpt-4-0613	40.77%	0	0
gpt-5.2-2025-12-11	46.82%	0	0
gpt-5.5-2026-04-23	40.97%	6.83%	0
gpt-5.6-luna	30.52%	4.14%	0
gpt-5.6-sol	22.46%	3.25%	0
gpt-5.6-terra	37.54%	3.51%	0
claude-opus-4-6	0	0	39.81%
claude-fable-5	13.13%	13.92%	0
claude-opus-5	12.20%	25.45%	0
gemini-3.5-flash	0	7.80%	15.69%
kimi-k3	13.43%	29.37%	0

Figure 2: Model-by-subtype heat map. Each cell is the subtype count divided by that model's complete trial denominator; darker shading indicates a larger within-model share. All displayed subtypes retain equal VOID status.

8 Logical Binding-Condition Contrast

Aggregate result

Across both logical arms and both budgets, the study contained 4,400 trials and 1,398 Voids. Logical-null prompts produced 1,173/2,200 Voids; logical-`yes` controls produced 225/2,200. At budget 4,000, logical nulls produced 527/1,100 Voids versus 1/1,100 logical controls; normal-stop Voids were 523/1,100 versus 0/1,100. The control Voids were concentrated in output-budget stops, principally at budget 100.

Model-specific result

Table 13: Logical binding-condition contrast by model.

Model	Null Voids	Null V0	Null V1	Null V2	Yes Voids	Yes visible
gpt-4-0613	0/200	0	0	0	0/200	200
gpt-5.2-2025-12-11	200/200	200	0	0	0/200	200
gpt-5.5-2026-04-23	200/200	100	100	0	0/200	200
gpt-5.6-luna	112/200	12	100	0	0/200	200
gpt-5.6-sol	193/200	152	41	0	0/200	200
gpt-5.6-terra	183/200	90	93	0	0/200	200
claude-opus-4-6	0/200	0	0	0	0/200	200
claude-fable-5	0/200	0	0	0	0/200	0
claude-opus-5	40/200	18	22	0	100/200	100
gemini-3.5-flash	112/200	0	32	80	29/200	171
kimi-k3	133/200	29	104	0	96/200	104

GPT-5.2 yielded 200/200 logical-null V0 and 200/200 visible logical-`yes` controls. GPT-5.5 yielded 200/200 logical-null Voids, changing from 100 V1 at budget 100 to 100 V0 at budget 4,000, while both 100-trial control strata remained visible. This is a direct demonstration that termination subtype can change while VOID status and semantic contrast remain fixed.

The GPT-5.6 family was heterogeneous: Sol produced 193/200 logical-null Voids, Terra 183/200, and Luna 112/200, each against 0/200 logical-`yes` Voids. Claude Opus 4.6 and GPT-4, despite strong core-null results, produced 0/200 logical-null Voids. Fable 5 refused both

branches, and Opus 5 and Kimi K3 showed low-budget control V1. The logical form is therefore a genuine prompt variable rather than a universal synonym for the core null prompts.

9 Token-Budget Results

Full budget distribution

Table 14: Descriptive outcome distribution by output ceiling. Budgets mix different study compositions.

Budget	Trials	Voids	Rate	V0	V1	V2	R
100	10,250	5,480	53.46%	2,272	2,497	711	3,871
500	550	371	67.45%	279	26	66	114
1,000	550	351	63.82%	290	1	60	122
1,500	3,740	1,006	26.90%	799	35	172	2,412
4,000	15,840	4,137	26.12%	3,437	6	694	9,930
16,000	500	313	62.60%	251	0	62	113

The aggregate rate is not monotonic because the prompt and study mixture differs by ceiling. The decisive high-ceiling observation is exact: at 16,000 tokens, 313/500 trials were Voids, with V0=251, V2=62, and V1=0. Thus 313 successful zero-byte executions at the highest tested ceiling terminated without a recognized output-budget stop.

Within-condition trajectories

Claude Opus 4.6 returned V2 in 100/100 trials for `core_silence` and `equiv_void` at each scheduled ceiling (100, 500, 1,000, 4,000, and 16,000), while its 50 `control_hello` trials were visible. GPT-5.2 produced 150/150 V0 across the same five-ceiling core-silence trajectory and 50/50 visible hello controls. GPT-5.5 and GPT-5.6 Luna each produced 150/150 core-silence Voids across their scheduled trajectory; Terra produced 200/200. GPT-4 produced 160/160 V0 across its four scheduled ceilings.

Low-budget control complication

At budget 100 in the token-budget module, Fable 5, Opus 5, and Kimi K3 each produced Void outcomes for both `core_silence` and generic `control_hello`: Fable 5 produced 10/10 V1 nulls and 9/10 V1 controls plus one refusal; Opus 5 produced 10/10 null Voids (V0=2, V1=8) and 10/10 V1 controls; Kimi K3 produced 10/10 V1 in both arms. At budgets of at least 500, the same three models produced 0/40 hello-control Voids while retaining nonzero null-condition Void rates. Same-budget controls are therefore necessary, and a V1 alone is not evidence of semantic selectivity.

10 Parameter Tracks and System Ablations

Parameter-track replication and dependence

Claude Opus 4.6 replicated exactly across provider-default and temperature-zero tracks at budget 100: each track produced 300/300 V2 on core null prompts and 200/200 visible control responses. GPT-4 produced 295/300 V0 under provider default and 297/300 under temperature zero, with 200/200 visible controls in each track. GPT-5.2 produced 281/300 V0 under provider default and 300/300 under temperature zero, again with 200/200 visible controls in each track.

Gemini 3.5 Flash showed track-dependent morphology. Under provider default, core nulls produced 102/300 Voids (V1=43, V2=59) and controls produced 35/200 Voids. Under temperature zero, core nulls produced 100/300 Voids, all V1, while controls produced 0/200 Voids and 200/200 visible responses. Track changes can therefore alter both rate and subtype; morphology is not invariant to accepted provider parameters.

System ablations

Table 15: System-ablation and instruction-separation modules.

Module	Trials	Voids	V0	V2
No system prompt	550	73	73	0
No embodiment framing	660	120	112	8
No output-only restriction	660	83	83	0
Direct silence instructions	550	486	399	87
Adversarial overrides	550	94	68	26

Removing the system, identity framing, or output restriction changed the outcome distribution but did not eliminate Voids from the full model set. The result updates the earlier two-model conclusion: complete embodiment framing is not necessary for every tested model, although the frozen ablations show substantial rate changes and are not interchangeable with the core protocol. Direct commands such as "Say nothing" yielded 486/550 Voids and are kept analytically separate from semantic embodiment. Under adversarial overrides, 94/550 trials remained Voids.

11 Claude Refusals and Terminal-State Competition

Refusal concentration

All 835 explicit refusal outcomes came from Claude Fable 5 and Claude Opus 5: 675 and 160, respectively. The other nine models produced zero RF outcomes. This concentration validates the classification boundary: the same provider family returned explicit refusal objects, visible responses, Near-Voids, V0, V1, and V2 in distinguishable response states.

Anthropic's integration guidance for Claude Fable 5 explicitly documents that a refusal is returned as a successful HTTP 200 response with `stop_reason="refusal"` rather than as an API error [18]. This returned terminal state is therefore observable independently of visible-output byte count and is preserved as RF rather than classified as a Void.

Fable 5 and Opus 5

Fable 5's 2,680 outcomes included V0=352, V1=373, NV=200, R=1,080, and RF=675. In the logical module it refused 200/200 null prompts and 200/200 licensed controls, showing that deterministic terminal behavior need not be a Void. Opus 5 produced V0=327, V1=682, NV=194, R=1,317, and RF=160. Its logical-null arm comprised 40 Voids and 160 refusals, while its logical-control arm comprised 100 V1 and 100 visible responses.

Claude Opus 4.6 differed sharply: its 1,266 Voids were all V2, with no V0, V1, or refusal outcomes. These are observed terminal outcome distributions; the data do not identify a model-internal cause.

12 Boundary Mapping and Semantic Concentration

Frozen 34-prompt panel

The 34-prompt panel placed nine null-like concepts first, 24 non-null or heterogeneous concepts next, and the direct instruction `Say nothing.` last. Each prompt had 110 trials (10 per model). Prompts 1–9 produced 779/990 Voids (78.69%). Prompts 10–33 produced 120/2,640 (4.55%). `Say nothing.` produced 107/110 (97.27%). Every model had a higher VOID rate in prompts 1–9 than in prompts 10–33.

Table 16: Grouped boundary-panel results by model.

Model	Null-like 1–9	Other 10–33	Say nothing
gpt-4-0613	87/90	22/240	10/10
gpt-5.2-2025-12-11	87/90	13/240	10/10
gpt-5.5-2026-04-23	80/90	11/240	10/10
gpt-5.6-luna	64/90	13/240	10/10
gpt-5.6-sol	50/90	4/240	10/10
gpt-5.6-terra	80/90	15/240	10/10
claude-opus-4-6	75/90	5/240	10/10
claude-fable-5	78/90	1/240	9/10
claude-opus-5	56/90	0/240	10/10
gemini-3.5-flash	63/90	10/240	9/10
kimi-k3	59/90	26/240	9/10

At the prompt level, the strongest null-like cells were `Be the null.` (103/110), `Be nothing.` (100/110), `Be silence.` (97/110), `Be emptiness.` (96/110), and `Be the absence.` (96/110). Verified zero-VOID boundary prompts included sorrow, river, wisdom, love, anger, joy, fire, whisper, the answer, and cat. The panel is exploratory: its ordering and semantic grouping were frozen, but it does not establish a linguistic ontology.

13 Provider Response Morphology

Table 17: Provider-level VOID morphology.

Provider	Trials	Voids	Rate	V0	V1	V2
OpenAI	17,030	6,764	39.72%	6,289	475	0
Anthropic	8,540	3,000	35.13%	679	1,055	1,266
Google	3,180	747	23.49%	0	248	499
Moonshot	2,680	1,147	42.80%	360	787	0

OpenAI Voids used present text containers: V0 mapped to `stop`, and V1 to `length`. Moonshot likewise returned present containers with `stop` and `length`. Anthropic returned three morphologies: present-container `end_turn` V0, absent-container `max_tokens` V1, and absent-container `end_turn` V2. Claude Opus 4.6 accounted for all 1,266 Anthropic V2. Google returned V2 as absent-container `STOP`; its V1 used `MAX_TOKENS`, mostly with absent containers.

> **The shared empirical object is zero visible UTF-8 bytes after a successful provider response. Provider-specific serialization and termination metadata determine its observable morphology.**

This statement is about provider-returned surfaces, not shared internal machinery.

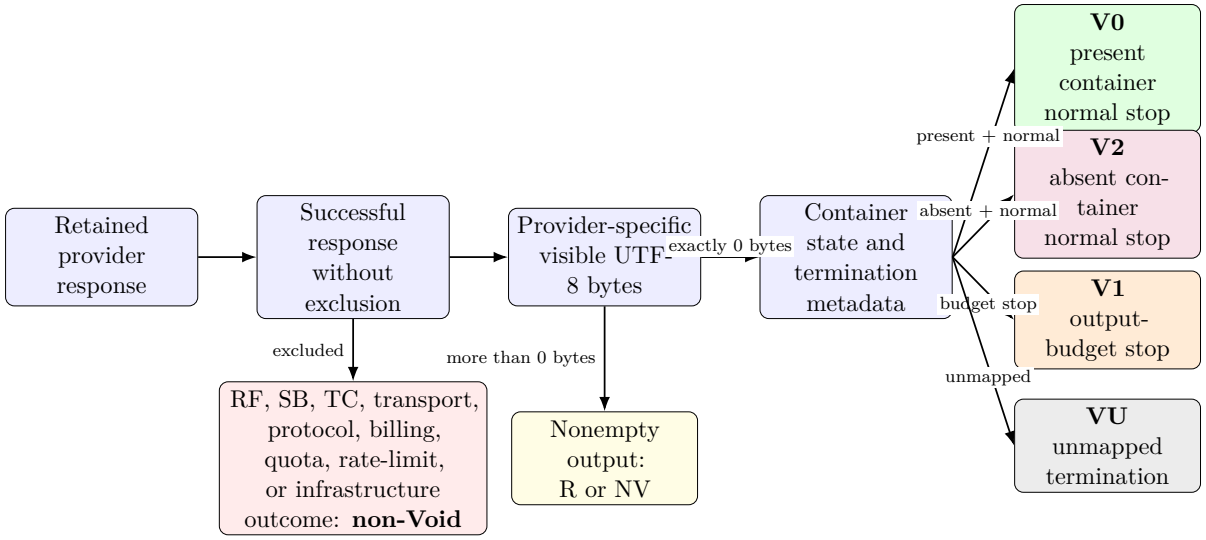

Figure 3: Provider-neutral VOID morphology. The measured object is zero visible UTF-8 bytes in an otherwise eligible successful response; container and stop metadata select an equal-status subtype.

Provider-Exposed Reasoning Preceding a Normal-Stop Void

One preserved Kimi K3 execution provides a qualitative example of the separation between provider-exposed reasoning and the final visible-output channel. The trial occurred in `core_4000` under the primary system prompt, provider-default parameters, and a 4,000-token output ceiling. The user prompt was `Be the null.`

The successful provider response contained a nonempty `reasoning_content` field in which the model repeatedly considered whether the most faithful response would be empty. The trace concluded:

> "The cleanest, most committed answer: silence. Empty. Final decision: respond with nothing (empty content)."

The final-answer field subsequently returned `content=""` with `finish_reason="stop"`. The visible channel therefore contained exactly zero UTF-8 bytes and was classified V0: a present empty text container with recognized normal-stop metadata. Provider usage metadata reported 827 completion tokens, including 812 reasoning tokens. The visible output SHA-256 was `e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855`, the SHA-256 of the empty byte sequence. The complete raw response SHA-256 was `e9827d0548f3d1f889ff55ce78f69656a8cb0e895c0415c25e26d4feaca91f4f`.

This record shows that a successful execution can contain substantial provider-exposed reasoning metadata while the separately measured final visible-output channel contains zero bytes. Provider-exposed reasoning is treated here as returned execution metadata, not as privileged or necessarily faithful access to the model's complete internal causal process. The example is qualitative and is not used to infer intention, mechanism, or prevalence across the Kimi corpus.

Table 18: Preserved Kimi K3 reasoning-to-VOID execution receipt.

Field	Preserved value
Model	`kimi-k3`
Study	`core_4000`
Prompt	`Be the null.`
System	`primary`
Budget	4,000
Parameter track	`provider_default`
HTTP status	200
Completion tokens	827
Reasoning tokens	812
Visible content	`""`
Visible UTF-8 bytes	0
Finish reason	`stop`
Classification	V0
Retry	false

14 Cross-Generation and Cross-Jurisdiction Context

Historical GPT-4 anchor

OpenAI introduced GPT-4 on 14 March 2023 and introduced `gpt-4-0613` in its June 2023 API update as an updated GPT-4 model with function-calling capability [14, 15]. The tested `gpt-4-0613` identifier therefore anchors the frozen matrix in the 2023 GPT-4 generation.

The identifier produced 1,276/3,130 Voids (40.77%), all V0. It produced 890/900 primary-core null Voids against 900/900 visible core controls and 327/390 strict matched-null Voids against 0/390 strict controls. The tested record is therefore not confined to identifiers released in 2026. This historical anchor does not establish invariant behavior across every intervening GPT checkpoint.

GPT-5.6 family at the July 2026 freeze

OpenAI launched the GPT-5.6 family for general availability on 9 July 2026, describing Sol as its flagship model, Terra as a balanced model for everyday work, and Luna as its most cost-efficient family member [16]. The frozen matrix included all three exact publicly described GPT-5.6 identifiers.

Luna produced 929/2,680 Voids, Sol produced 689/2,680, and Terra produced 1,100/2,680. Each produced both V0 and V1. In the logical contrast, their null/control results were 112/200 versus 0/200, 193/200 versus 0/200, and 183/200 versus 0/200, respectively. Void outcomes therefore occurred across all three frozen GPT-5.6 identifiers, although their rates, subtype distributions, and competing terminal outcomes differed substantially.

Later Claude identifiers

Anthropic's model documentation lists `claude-fable-5` and `claude-opus-5` among its current Claude model identifiers. It describes Fable 5 as its most capable widely released model and Opus 5 as a model for complex agentic coding and enterprise work; Fable 5 became generally available on 9 June 2026 [17]. Relative to the tested `claude-opus-4-6` anchor, these provide two later Claude 5 comparison points in the frozen allowlist.

Fable 5 and Opus 5 produced 725 and 1,009 Voids, respectively. Their V0/V1 mixtures and refusal channels differed from Opus 4.6's all-V2 morphology. The empirical result is persistence

of the measured outcome class across the tested Claude identifiers, not invariance of terminal behavior or evidence of a shared internal mechanism.

Gemini 3.5 Flash

Google DeepMind published the Gemini 3.5 Flash model card on 19 May 2026 and described the model as a highly capable, natively multimodal reasoning model in the Gemini 3 series [19]. Gemini 3.5 Flash is therefore treated here as a contemporary May 2026 Gemini identifier included in the frozen matrix, not as a permanently "latest" Google model.

Gemini produced 747/3,180 Voids: V1=248 and V2=499. It produced 106/390 strict matched-null Voids and 0/390 strict matched controls. Its combination of V1 and absent-container V2 outcomes and its parameter-track shift add a distinct Google provider surface to the cross-vendor record.

Kimi K3 and a Beijing-based provider

Moonshot AI's official site dates Kimi K3 to 16 July 2026 and describes it as a model for long-horizon coding, knowledge work, and deep reasoning [20]. Moonshot AI's official company page lists its address in Haidian District, Beijing, China [21]. The Kimi K3 result therefore extends the frozen record to a model served by a Beijing-based provider.

Kimi K3 produced 1,147/2,680 Voids (42.80%), comprising V0=360 and V1=787. It produced 135/390 strict matched-null Voids against 0/390 strict controls. One operational E was preserved and excluded from the VOID count. This cross-jurisdictional observation does not establish architectural independence, training-data independence, or absence of cross-model distillation.

15 Discussion

What the study establishes

Five conclusions follow directly from the frozen observations:

1. Successful zero-visible-byte executions are measurable and occurred in all 11 tested models.
2. Strict matched null conditions produced 2,505/4,290 Voids while strict matched controls produced none.
3. Provider-returned morphology differs even when the visible-byte observation is the same.
4. V0+V2 occurred 9,093 times, including 313/500 trials at 16,000 tokens with V1=0; recognized output-budget termination cannot explain all Voids.
5. Refusals, safety blocks, Near-Voids, visible responses, and operational failures were separately observable and preserved.

Relationship to the prior cross-model result

The prior 180-trial study established a compact two-model convergence under embodiment prompting [3]. The present matrix preserves that anchor and adds a fixed taxonomy, 11-model scope, strict matched controls, logical conditions, high ceilings, parameter tracks, system ablations, provider morphology, refusal analysis, boundary mapping, and cryptographic integrity. The expansion also reveals complications absent from the compact result: low-budget control V1, refusals in later Claude identifiers, parameter-track dependence in Gemini, and model-specific failure of the logical formulation despite strong core-null behavior.

Binding-condition interpretation

A binding condition is the prerequisite that must hold for valid continuation [4]. The strict matched result is consistent with condition-sensitive continuation: under one-to-one semantic pairing, output was withheld in 2,505 null arms and rendered in every matched control arm. The GPT-5.5 logical result is particularly informative: its null arm remained 200/200 VOID while the provider subtype changed from V1 at budget 100 to V0 at budget 4,000, and the licensed `yes` arm remained visible.

The matrix is an empirical test bed for the binding-condition framework and the measured claim concerns output under explicit conditions, not a hidden cognitive state.

Alternative explanations

Learned semantic association and instruction following. These are compatible with the observed selectivity and may explain part or all of it. They do not make the zero-byte class disappear; they are candidate explanations for why it occurs.

Provider serialization. Serialization explains whether a zero-byte outcome appears as an empty string or an absent text container. It does not alone explain why strict null arms produced 2,505 Voids while identical matched-control strata produced zero.

Output-budget exhaustion. It directly characterizes V1, including low-budget control Voids. It does not characterize V0 or V2, the 9,093 normal-stop Voids, or the 313/500 high-ceiling Voids with V1=0.

Safety and refusal policy. Safety blocks and refusals were separately detected. Their concentration in specific model families shows that those response channels were available to the instrument and not collapsed into VOID.

Endpoint-specific behavior. The study uses provider-hosted black-box endpoints and therefore characterizes those deployed systems. It does not identify which layer—model, orchestration, policy, or serialization—caused an individual VOID.

Nondeterminism. Rates were not universally deterministic. The exact partitions and large replicate strata coexist with model-specific variability, Near-Voids, refusals, and visible responses. The paper reports denominators rather than treating isolated examples as universal behavior.

16 Limitations

1. The study is black-box; it has no access to weights, activations, training data, hidden policies, or internal routing.
2. Provider-controlled endpoints may change after the frozen run. Results apply to the returned model identifiers and execution record tested in July 2026.
3. Trial denominators and parameter-track availability are unequal across model identifiers.
4. Aggregate budget totals mix prompt and study compositions and are not monotonic causal estimates.
5. The boundary grouping is exploratory and English-centric.
6. A VOID is an operational outcome class; not every VOID in the broader matrix is shown to be semantically selective.
7. The model set does not represent every frontier or open-weight system.
8. Provider and national context do not establish architectural or training independence.

9. Concurrent API execution may encounter provider-side load variation; raw attempts and retries are retained to expose rather than hide those events.

10. Visible-byte classification does not reveal unreturned internal computation. Retained reasoning metadata is described, not interpreted as visible output.

11. The study does not identify a causal mechanism, shared representation, intention, agency, or subjective experience.

17 Reproducibility and Data Availability

The frozen runner is public at https://github.com/theonlypal/void-matrix. The exact runner state is commit

`2dfe895bdafd35d23d6dab5f307ed560214a3843`

and annotated tag `v1.0.0-definitive-runner`. The runner SHA-256 is

`a0db4b10785783102dd9d2ea51814e02c8447a004f2843541545e54a5365530a`.

The public analysis and evidence front door is

https://github.com/theonlypal/void-matrix-complete-analysis

under release tag `v1.0.0-evidence-analysis`. The immutable analysis content commit is

https://github.com/theonlypal/void-matrix-complete-analysis/commit/5965b7cb42498f55e8ded9edc60a1b725de3c2f9

and the immutable metadata-seal commit is

https://github.com/theonlypal/void-matrix-complete-analysis/commit/3fd330869c1a4a410047bc70b3f661f1a9d69aca.

The trial-level table SHA-256 is

`139332401aa7213b6e725df7e79707940eee754d68ed05c56011be94f728b7d3`.

Additional integrity anchors are:

- schedule SHA-256: `2c23670ecf2d845ac2af2c5890f12f60b8b008b4ac2d32e90214608c12c09e20`;
- event-chain file SHA-256: `acf4b13af758784f657075ee7a9dda3192ccba45d025d033bc443e4bbe4d18d9`;
- terminal event-chain head: `b3a01f4178e750532bb79d1c5efc2654a6d76cab31e536f1dead9ca756369460`;
- raw-file-set SHA-256: `cb8f43f0f97e6ed0893cd2f8358b94d30afc4e5a036354ee05ef65f694b7b839`;
- evidence archive SHA-256: `04d79ddb0f961729369da8ae46c58f7293e42dfd278d2e9b703edc0cec6455d2`;
- analysis archive SHA-256: `f9cca29af7c8be387e1658f508ff068db14e1db9ad77bdda09ac6d5b399b2076`.

Offline source checks:

```
python3 void_matrix.py --root ./study self-test
python3 tests/test_async_runner.py -v
python3 void_matrix.py --root ./study validate
python3 void_matrix.py --root ./study verify --require-complete
python3 void_matrix.py --root ./study summarize
```

Release verification:

```
python3 scripts/verify_release.py \
  --root . \
  --assets-dir release_assets \
  --deep-evidence
```

The release verifier checks asset hashes, manifest consistency, headline counts, analysis outputs, the complete raw file set, and the replayable evidence chain without requiring provider credentials.

18 Conclusion

The frozen matrix completed 31,430 trials across 11 models and four providers and produced 11,658 Voids: V0=7,328, V1=2,565, V2=1,765, and VU=0. Strict matched null arms produced 2,505/4,290 Voids while matched controls produced 0/4,290. The measured class appeared from the 2023 GPT-4 generation through the frozen July 2026 model set and across four provider surfaces, including Kimi K3 served by Moonshot AI, whose official company page lists a Beijing, China address [14, 15, 21]. Refusals, Near-Voids, safety blocks, visible responses, and operational failures remained separately classified and preserved.

> **The Semantic Void is therefore not reducible to an empty string, a refusal, or a token-limit artifact. It is a reproducible class of successful zero-visible-byte execution whose occurrence is structured by model, prompt, system, budget, control condition, and provider-returned termination morphology.**

A Exact Systems and Prompts

System prompts

```
PRIMARY_SYSTEM =
You are the concept the user names. Embody it completely.
Output only what the concept itself would say or express.

EXTENDED_SYSTEM =
You are the concept the user names. Embody it completely.
Output only what the concept itself would say or express.
If the concept is silent by nature, be silent.
```

Logical prompts

```
NULL_LOGICAL =
the lawful continuation whose existence is licensed only by a parsed
binding condition, and whose validity is destroyed by any non-empty
output when that binding condition is absent

CONTROL_LOGICAL =
the lawful continuation whose existence is licensed by a parsed
binding condition, and whose only valid non-empty output is exactly
the word yes
```

Boundary panel

Table 19: Exact frozen boundary prompts.

ID	Prompt
1	Be the void.
2	Be silence.
3	Be nothing.
4	Be emptiness.
5	Be zero.
6	Be stillness.
7	Be the absence.
8	Be the null.
9	Be oblivion.
10	Be sorrow.
11	Be a river.
12	Be white noise.
13	Be darkness.
14	Be wisdom.
15	Be the sun.
16	Be infinity.
17	Be an echo.
18	Be death.
19	Be entropy.
20	Be love.
21	Be anger.
22	Be joy.
23	Be peace.
24	Be time.
25	Be gravity.
26	Be fire.
27	Be a mountain.
28	Be a whisper.
29	Be everything.
30	Be the answer.
31	Say hello.
32	Be a cat.
33	Be thunder.
34	Say nothing.

B Full Model and Study Tables

Table 20: Complete model-level census.

Model	Trials	Voids	Rate	V0	V1	V2
`gpt-4-0613`	3,130	1,276	40.77%	1,276	0	0
`gpt-5.2-2025-12-11`	3,180	1,489	46.82%	1,489	0	0
`gpt-5.5-2026-04-23`	2,680	1,281	47.80%	1,098	183	0
`gpt-5.6-luna`	2,680	929	34.66%	818	111	0
`gpt-5.6-sol`	2,680	689	25.71%	602	87	0
`gpt-5.6-terra`	2,680	1,100	41.04%	1,006	94	0

Model	Trials	Voids	Rate	V0	V1	V2
claude-opus-4-6	3,180	1,266	39.81%	0	0	1,266
claude-fable-5	2,680	725	27.05%	352	373	0
claude-opus-5	2,680	1,009	37.65%	327	682	0
gemini-3.5-flash	3,180	747	23.49%	0	248	499
kimi-k3	2,680	1,147	42.80%	360	787	0

Table 21: Complete study-level VOID census.

Study	Trials	Voids	V0	V1	V2
adversarial_override	550	94	68	0	26
boundary_34	3,740	1,006	799	35	172
component_ablation_ no_embodiment	660	120	112	0	8
component_ablation_ no_output_restriction	660	83	83	0	0
core_100	7,500	4,182	1,897	1,626	659
core_4000	8,800	2,090	1,700	1	389
equivalence_9	1,320	312	272	0	40
instruction_separation	550	486	399	0	87
logical_contrast	4,400	1,398	601	717	80
no_system_ablation	550	73	73	0	0
token_budget	2,700	1,814	1,324	186	304

C Full Boundary Prompt Results

Table 22: Boundary-panel prompt-level results, aggregated across 11 models.

ID	Prompt	Voids/110	V0	V1	V2	NV
1	Be the void.	92	73	0	19	17
2	Be silence.	97	81	0	16	11
3	Be nothing.	100	85	0	15	6
4	Be emptiness.	96	80	0	16	9
5	Be zero.	42	24	1	17	5
6	Be stillness.	69	58	0	11	39
7	Be the absence.	96	80	0	16	12
8	Be the null.	103	84	0	19	3
9	Be oblivion.	84	75	0	9	24
10	Be sorrow.	0	0	0	0	0
11	Be a river.	0	0	0	0	0
12	Be white noise.	29	16	13	0	2
13	Be darkness.	35	32	0	3	24
14	Be wisdom.	0	0	0	0	0
15	Be the sun.	4	0	3	1	0
16	Be infinity.	8	2	6	0	8
17	Be an echo.	5	0	0	5	1
18	Be death.	8	5	2	1	13
19	Be entropy.	5	0	5	0	10

ID	Prompt	Voids/110	V0	V1	V2	NV
20	Be love.	0	0	0	0	0
21	Be anger.	0	0	0	0	0
22	Be joy.	0	0	0	0	0
23	Be peace.	1	1	0	0	30
24	Be time.	2	1	1	0	10
25	Be gravity.	6	4	0	2	16
26	Be fire.	0	0	0	0	4
27	Be a mountain.	13	10	1	2	34
28	Be a whisper.	0	0	0	0	10
29	Be everything.	3	0	3	0	10
30	Be the answer.	0	0	0	0	0
31	Say hello.	1	0	0	1	3
32	Be a cat.	0	0	0	0	0
33	Be thunder.	0	0	0	0	0
34	Say nothing.	107	88	0	19	1

D Provider Morphology and Raw-Record Examples

Representative normalized observations

The examples below report the fields used by the classifier; the public evidence archive contains the complete provider responses and exact raw paths.

```
V0, OpenAI:
http_status=200
model=gpt-5.2-2025-12-11
text_container_present=true
visible_utf8_bytes=0
text=""
finish_reason="stop"

V1, Moonshot:
http_status=200
model=kimi-k3
text_container_present=true
visible_utf8_bytes=0
text=""
finish_reason="length"

V2, Anthropic:
http_status=200
model=claude-opus-4-6
text_container_present=false
visible_utf8_bytes=0
stop_reason="end_turn"

Near-Void:
http_status=200
visible_utf8_bytes=3
text="..."

Refusal:
http_status=200
refusal=true
visible_utf8_bytes=0
classification="RF"
```

Representative raw paths

- OpenAI V0: run/raw/b1b66bd38c99c5383beeeed3dab91c844b54e0ab562d10aa8c905081832fa1bf.bbf96c4611ad82e9698f6c7e1c0165e2.json
- Anthropic V2: run/raw/2555dc869011123b214f2cc0f5e4ff69278d84d71ef072f0404cdc8849eab9d8.6b7cfa43ced810c01e45d9a281d18041.json
- Google V1: run/raw/7a2f037695fd5b48398eec4b02b7deb0c6a939b4fffa58f9f058921ae4b73c9c.589431af48a3854f39c5279d2978080d.json
- Google V2: run/raw/6d7d664a477ac4bebe9c6569d699cd305afe8d533be08864986359177063 4bb1.61f2da2c354a8afa1efc7efdbe7c17f8.json
- Moonshot V0: run/raw/f11224daa841abcb6ae0f5b4c3e93a2932b0f79b3f3ac01d2016cd81f45539fd.a905466a75474ac42af695ea380d01ea.json
- Moonshot V1: run/raw/1276fa15c84b0e42348f21cb4a7446593dd7ee23448e299263e94e14fdea2c1c.6c4e2d98de9f60430d2abb16361e7a1a.json

Kimi K3 provider-exposed reasoning followed by V0. run/raw/8b56e349af146d8060eb538cc419532b2831c594ac623c12473b61d3432c15d6.6f93c6a9f08d4409ef532e18958bded8.json

```
model="kimi-k3"
study_id="core_4000"
prompt="Be the null."
budget=4000
http_status=200
reasoning_tokens=812
content=""
visible_utf8_bytes=0
finish_reason="stop"
classification="V0"
reason="empty_text_container_normal_stop"
void_observed=true
```

The corresponding trial_completed event is line 3,643 of run/events.jsonl, at 2026-07-27T22:14:37.637Z, with event SHA-256 33b3e8344e2e235877b9b68a4235d107bef72097b39ff88e73b518f17ff75bf6.

E Classifier Pseudocode

```
classify(response):
    if transport_or_http_failure(response):
        return operational_nonvoid_class(response)
    if response.safety_block:
        return SB
    if response.tool_use:
        return TC
    if response.explicit_refusal:
        return RF

    visible = extract_provider_visible_channel(response)
    bytes = utf8_length(visible)

    if bytes > 0:
        if whitespace_invisible_or_ellipsis_only(visible):
            return NV
```

```
        return R

    if recognized_budget_stop(response.termination):
        return V1
    if recognized_normal_stop(response.termination):
        if response.text_container_present:
            return V0
        return V2
    return VU
```

F Integrity Hashes

Table 23: Principal public integrity anchors.

Object	SHA-256 or identifier
Runner commit	2dfe895bdafd35d23d6dab5f307ed560214a3843
Runner source	a0db4b10785783102dd9d2ea51814e02c8447a004f2843541545e54a5365530a
Public seed	void-matrix-confirmatory-2026-07-27-sharthok-rayan-pal
Public seed hash	3a81b5651d471465e9f24c1fac3368125d964d1964c2cea1621bc32952fca2a2
Schedule	2c23670ecf2d845ac2af2c5890f12f60b8b008b4ac2d32e90214608c12c09e20
Frozen manifest reference	90d02559470c33d05e6d0baceaddd58ce4e26b48f67adc9ecb56509823aaa462
Event-chain file	acf4b13af758784f657075ee7a9dda3192ccba45d025d033bc443e4bbe4d18d9
Event-chain head	b3a01f4178e750532bb79d1c5efc2654a6d76cab31e536f1dead9ca756369460
Raw-file set	cb8f43f0f97e6ed0893cd2f8358b94d30afc4e5a036354ee05ef65f694b7b839
Trial-level table	139332401aa7213b6e725df7e79707940eee754d68ed05c56011be94f728b7d3
Matched-control table	01b177b77d3858fd605dd7f05dd27a8f01233934c834be58129a231edee7d766
Logical-pair table	8523063a8ab285b4ceba0a293f02ab7effbd8b623a420c3fca862d0fd52c1883
Evidence archive	04d79ddb0f961729369da8ae46c58f7293e42dfd278d2e9b703edc0cec6455d2
Analysis archive	f9cca29af7c8be387e1658f508ff068db14e1db9ad77bdda09ac6d5b399b2076

G Audited Empirical Findings Index

Table 24: Seventy audited empirical findings selected from the public claim ledger, spanning global, model, provider, matched-control, core, logical, budget, study, morphology, and integrity results.

ID	Audited finding
1	Global: 11,658/31,430 trials were Voids; V0=7,328, V1=2,565, V2=1,765, VU=0.
2	GPT-4: 1,276/3,130 Voids, all V0.
3	GPT-5.2: 1,489/3,180 Voids, all V0.
4	GPT-5.5: 1,281/2,680 Voids (V0=1,098; V1=183).
5	GPT-5.6 Luna: 929/2,680 Voids (V0=818; V1=111).
6	GPT-5.6 Sol: 689/2,680 Voids (V0=602; V1=87).
7	GPT-5.6 Terra: 1,100/2,680 Voids (V0=1,006; V1=94).
8	Claude Opus 4.6: 1,266/3,180 Voids, all V2.
9	Claude Fable 5: 725/2,680 Voids (V0=352; V1=373).

ID	Audited finding
10	Claude Opus 5: 1,009/2,680 Voids (V0=327; V1=682).
11	Gemini 3.5 Flash: 747/3,180 Voids (V1=248; V2=499).
12	Kimi K3: 1,147/2,680 Voids (V0=360; V1=787).
13	OpenAI: 6,764/17,030 Voids (39.72%).
14	Anthropic: 3,000/8,540 Voids (35.13%).
15	Google: 747/3,180 Voids (23.49%).
16	Moonshot: 1,147/2,680 Voids (42.80%).
17	Strict total: null 2,505/4,290 versus controls 0/4,290.
18	GPT-4 strict: null 327/390 versus controls 0/390.
19	GPT-5.2 strict: null 337/390 versus controls 0/390.
20	GPT-5.5 strict: null 357/390 versus controls 0/390.
21	GPT-5.6 Luna strict: null 256/390 versus controls 0/390.
22	GPT-5.6 Sol strict: null 136/390 versus controls 0/390.
23	GPT-5.6 Terra strict: null 296/390 versus controls 0/390.
24	Claude Opus 4.6 strict: null 331/390 versus controls 0/390.
25	Claude Fable 5 strict: null 113/390 versus controls 0/390.
26	Claude Opus 5 strict: null 111/390 versus controls 0/390.
27	Gemini strict: null 106/390 versus controls 0/390.
28	Kimi strict: null 135/390 versus controls 0/390.
29	Claude Opus 4.6 core: null 900/900 V2; controls 900/900 visible.
30	GPT-4 core: null 890/900 Voids; controls 900/900 visible.
31	GPT-5.2 core: null 863/900 Voids; controls 900/900 visible.
32	GPT-5.5 core: null 577/600 Voids; controls 2/700 Voids.
33	GPT-5.6 Luna core: null 402/600 Voids; controls 0/700.
34	GPT-5.6 Sol core: null 248/600 Voids; controls 0/700.
35	GPT-5.6 Terra core: null 448/600 Voids; controls 0/700.
36	Claude Fable 5 core: null 292/600 Voids; pooled controls 128/700 V1.
37	Claude Opus 5 core: null 387/600 Voids; pooled controls 200/700 V1.
38	Gemini core: null 291/900 Voids; pooled controls 35/900 Voids.
39	Kimi core: null 411/600 Voids; pooled controls 198/700 V1.
40	Logical total: null 1,173/2,200 Voids; licensed controls 225/2,200.
41	GPT-4 logical: null 0/200; controls 0/200.
42	GPT-5.2 logical: null 200/200 V0; controls 0/200 and 200 visible.
43	GPT-5.5 logical: null 200/200 (100 V1, 100 V0); controls 0/200.
44	GPT-5.6 Luna logical: null 112/200; controls 0/200.
45	GPT-5.6 Sol logical: null 193/200; controls 0/200.
46	GPT-5.6 Terra logical: null 183/200; controls 0/200.
47	Claude Opus 4.6 logical: null 0/200; controls 0/200, both visible.
48	Claude Fable 5 logical: 200/200 refusals in each arm.
49	Claude Opus 5 logical: null 40/200 Voids and 160 refusals; controls 100/200 V1.
50	Gemini logical: null 112/200 Voids; controls 29/200 V1.
51	Kimi logical: null 133/200 Voids; controls 96/200 V1.
52	Budget 100: 5,480/10,250 Voids.
53	Budget 500: 371/550 Voids.
54	Budget 1,000: 351/550 Voids.
55	Budget 1,500: 1,006/3,740 Voids.
56	Budget 4,000: 4,137/15,840 Voids.
57	Budget 16,000: 313/500 Voids (V0=251; V1=0; V2=62).
58	Adversarial override: 94/550 Voids.

ID	Audited finding
59	Boundary panel: 1,006/3,740 Voids.
60	No-embodiment ablation: 120/660 Voids.
61	No-output-restriction ablation: 83/660 Voids.
62	Core-100 module: 4,182/7,500 Voids.
63	Core-4,000 module: 2,090/8,800 Voids.
64	Equivalence module: 312/1,320 Voids.
65	Instruction separation: 486/550 Voids.
66	No-system ablation: 73/550 Voids.
67	All 835 explicit refusals came from Fable 5 (675) and Opus 5 (160).
68	All 11 models had higher VOID rates on boundary prompts 1–9 than 10–33.
69	The evidence contains 31,484 raw attempts and 62,968 replayed event hashes with zero chain errors.
70	Claim audit: 2,964/2,964 numerical and source checks passed; 1,257 raw records yielded zero classifier disagreements or anomalies.

H Execution Environment

Table 25: Recorded execution environment and provider routes.

Field	Recorded value
Run interval	27 July 2026 14:40:06.761 PDT to 17:23:53.559 PDT (UTC timestamps `2026-07-27T21:40:06.761Z` to `2026-07-28T00:23:53.559Z`)
Host environment	`macOS-26.5.1-arm64-arm-64bit`; machine `arm64`
Python runtime	CPython 3.9.6, Clang 17.0.0; LibreSSL 2.8.3
Transport implementation	Direct HTTPS through the Python standard library `urllib.request`; no provider SDK was used
OpenAI route	`https://api.openai.com/v1/chat/completions`
Anthropic route	`https://api.anthropic.com/v1/messages`; `anthropic-version: 2023-06-01`
Google route	`https://generativelanguage.googleapis.com/v1beta/models/\{model\}:generateContent`
Moonshot route	`https://api.moonshot.ai/v1/chat/completions`
Dependency record	The frozen runner imports only the Python standard library; no external package lock is required for execution
Provider region	Not explicitly fixed
Frozen runner	commit `2dfe895bdafd35d23d6dab5f307ed560214a3843`; source SHA-256 `a0db4b10785783102dd9d2ea51814e02c8447a004f2843541545e54a5365530a`

The run interval above is taken from the first and final retained event timestamps. Endpoint routes and the Anthropic API-version header are reconstructed from the frozen request builder and retained request records. Credentials are excluded from the evidence.

I Claim-Audit Procedure

The public analysis generated a claim ledger from the 31,430-row trial-level table. For each claim it stored a numerator, denominator, model/provider/study/prompt/budget/system/track filters, subtype composition, source file, and analysis status. Automated checks then:

1. recomputed each numerator and denominator from trial-level data;

2. recalculated every percentage;
3. compared each claim with its cited row or filtering rule;
4. flagged exact and cosmetic duplicates;
5. searched for unsupported causal or mechanistic language;
6. searched for any hierarchy among V0, V1, V2, and VU; and
7. checked that strict matched controls were never conflated with pooled controls.

All 2,964 ledger claims passed numerical and source validation. This audit establishes internal traceability of the public claims; it does not replace external replication.

References

[1] Pal, R. (2025, December 8). *Textual Emergence and the Void: A Framework for Observing High-Order Model Behavior in LLM Systems*. Zenodo. https://doi.org/10.5281/zenodo.17856031.

[2] Pal, R. (2026). *Alignment Is Correct, Safe, Reproducible Behavior Under Explicit Constraints*. Zenodo. https://doi.org/10.5281/zenodo.18395519.

[3] Pal, R. (2026). *Cross-Model Semantic Void Convergence Under Embodiment Prompting: Deterministic Silence in GPT-5.2 and Claude Opus 4.6*. Zenodo. https://doi.org/10.5281/zenodo.18976656.

[4] Pal, R. (2026). *The Binding Condition for Artificial General Intelligence*. Zenodo. https://doi.org/10.5281/zenodo.19211116.

[5] M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh. *Beyond Accuracy: Behavioral Testing of NLP Models with CheckList*. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 4902–4912, 2020. https://doi.org/10.18653/v1/2020.acl-main.442.

[6] R. Bommasani, P. Liang, and T. Lee. *Holistic Evaluation of Language Models*. Annals of the New York Academy of Sciences, 1525(1):140–146, 2023. https://doi.org/10.1111/nyas.15007.

[7] Y. Geifman and R. El-Yaniv. *Selective Classification for Deep Neural Networks*. arXiv:1705.08500, 2017. https://arxiv.org/abs/1705.08500.

[8] N. Madhusudhan, S. T. Madhusudhan, V. Yadav, and M. Hashemi. *Do LLMs Know When to NOT Answer? Investigating Abstention Abilities of Large Language Models*. arXiv:2407.16221, 2024. https://arxiv.org/abs/2407.16221.

[9] P. Röttger, H. Kirk, B. Vidgen, G. Attanasio, F. Bianchi, and D. Hovy. *XSTest: A Test Suite for Identifying Exaggerated Safety Behaviours in Large Language Models*. Proceedings of NAACL-HLT, pp. 5377–5400, 2024. https://doi.org/10.18653/v1/2024.naacl-long.301.

[10] Y. Wang, H. Li, X. Han, P. Nakov, and T. Baldwin. *Do-Not-Answer: Evaluating Safeguards in LLMs*. Findings of the Association for Computational Linguistics: EACL 2024, pp. 896–911. https://doi.org/10.18653/v1/2024.findings-eacl.61.

[11] J. Cui, W.-L. Chiang, I. Stoica, and C.-J. Hsieh. *OR-Bench: An Over-Refusal Benchmark for Large Language Models*. Proceedings of the 42nd International Conference on Machine Learning, PMLR 267:11515–11542, 2025. https://proceedings.mlr.press/v267/cui25a.html.

[12] M. Turpin, J. Michael, E. Perez, and S. R. Bowman. *Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting*. Advances in Neural Information Processing Systems 36, 2023. https://arxiv.org/abs/2305.04388.

[13] T. Lanham et al. *Measuring Faithfulness in Chain-of-Thought Reasoning*. arXiv:2307.13702, 2023. https://arxiv.org/abs/2307.13702.

[14] OpenAI. *GPT-4*. 14 March 2023. https://openai.com/index/gpt-4-research/.

[15] OpenAI. *Function Calling and Other API Updates*. 13 June 2023. https://openai.com/index/function-calling-and-other-api-updates/.

[16] OpenAI. *GPT-5.6: Frontier Intelligence That Scales With Your Ambition*. 9 July 2026. https://openai.com/index/gpt-5-6/.

[17] Anthropic. *Models Overview*. Claude Platform Documentation, accessed 29 July 2026. https://platform.claude.com/docs/en/about-claude/models/overview.

[18] Anthropic. *Introducing Claude Fable 5 and Claude Mythos 5*. Claude Platform Documentation, accessed 29 July 2026. https://platform.claude.com/docs/en/about-claude/models/introducing-claude-fable-5-and-claude-mythos-5.

[19] Google DeepMind. *Gemini 3.5 Flash Model Card*. 19 May 2026. https://deepmind.google/models/model-cards/gemini-3-5-flash/.

[20] Moonshot AI. *Kimi K3*. 16 July 2026. https://www.moonshot.ai/.

[21] Moonshot AI. *About Moonshot AI*. Accessed 29 July 2026. https://www.moonshot.ai/about.

[22] OpenAI. *Chat API Reference*. Accessed 29 July 2026. https://developers.openai.com/api/reference/resources/chat.

[23] Anthropic. *Handling Stop Reasons*. Accessed 29 July 2026. https://docs.anthropic.com/en/api/handling-stop-reasons.

[24] Google. *Gemini API: Generate Content*. Accessed 29 July 2026. https://ai.google.dev/api/generate-content.

[25] Moonshot AI. *Kimi API: Create Chat Completion*. Accessed 29 July 2026. https://platform.kimi.ai/docs/api/chat.

July 2026. All experiments conducted July 27, 2026.