METEHAN.AI RESEARCH REPORT · PREPRINT FIELD GUIDE

AI Visibility Data Collection: A Field Glossary and Audit Map

Metehan Yesilyurt

AI Search & Information Retrieval Researcher · metehan.ai

https://metehan.ai/pdf/ai-visibility-data-collection-field-glossary.pdf

Abstract. This companion document maps the vocabulary and decision points behind AI visibility data collection. It gives you a glossary, a concept map, audit steps, and worked examples so you can judge whether a tool's numbers reflect real consumer AI answers. I focus on criteria and tests, not vendor narrative.

Keywords: AI visibility, data collection, API vs UI, sampling, methodology transparency, vendor audit

Companion to the article How AI Visibility Tools Actually Collect Data: API vs UI Scraping, Sampling and Methodology Transparency Explained. The article covers the topic in full; this document is a separate, practical field glossary and concept map.

1. How I Use This Glossary

I treat AI visibility data collection as a supply chain: prompts go in, observations come out, and metrics are derived from those observations. If you cannot trace a metric back to the individual answers behind it, you cannot audit it. This document is a field guide for that traceability problem. Use it when you evaluate tools, write internal requirements, or explain to stakeholders why two dashboards disagree. The source article [1] gives the narrative and vendor questions. Here I give the vocabulary, decision rules, and tests. I assume you already accept that AI answers vary. My goal is to help you separate measurement from approximation. I use this reference card in three situations: when a vendor sends a methodology page, when a stakeholder asks why numbers moved, and when I need to write acceptance criteria for a tracking tool. The glossary defines terms so that a sales call cannot redefine them. The decision rules give me a default. The tests give me a way to falsify a claim. The worked examples show how the pieces fit.

2. Concept Map: Entities and Relationships

A concept map is more useful than a feature list because it shows where a vendor can hide a gap. The core entities are prompts, channels, collection methods, runs, observations, extractions, aggregates, and annotations. The relationships are strict: a channel is accessed through a collection method; a collection method produces runs; a run yields an observation; an observation is parsed into mentions and citations; mentions and citations are aggregated into metrics; a sampling plan governs how many runs exist; a model version and method change log annotate the time series. Raw answer access is the edge that connects a metric back to its observation. Remove that edge and the map becomes a black box.

- Prompt: the input question or phrase you track. A prompt set is a deliberate sample, not the whole market.

- Channel: the consumer surface, such as ChatGPT, Perplexity, Google AI Overviews, AI Mode, or Gemini [1].
- Collection method: the mechanism used to obtain answers, either API calls, real UI execution, or a hybrid [1].
- Run: one execution of one prompt on one channel at one time.
- Observation: the full rendered answer, including text, sources, and interface modules.
- Extraction: the rules that turn an observation into a mention, citation, or position.
- Aggregate: the metric shown on a dashboard, such as a visibility score or share of voice.
- Annotation: the metadata that explains model updates or method changes.
- Raw answer access: the ability to inspect the observation behind an aggregate [1].

The map also has a direction. Prompts flow into channels, channels into collection methods, collection methods into runs, runs into observations, observations into extractions, and extractions into aggregates. Annotations sit above the time series. When a vendor says they track visibility, I ask which edge they are describing. If they cannot name the edge, they are describing a feeling. If they describe an edge but cannot show the data that travels along it, they are describing a black box. The concept map is my first diagnostic.

3. Field Glossary

The glossary is not a dictionary for its own sake. Each term is a place where a vendor can be vague. I want a working definition that can be tested. I keep this glossary close when reading vendor documentation. The definitions are operational: they tell you what to ask for, not what to believe.

Term	Working definition	Why it matters
API collection	Sending prompts to a model provider's developer API and parsing the response [1].	Fast and cheap, but may not match the consumer product's model, retrieval, or interface [1].
UI execution	Running prompts through the real consumer interface and recording the rendered answer [1].	Captures production model, live retrieval, citations, and interface modules [1].
Hybrid collection	Using API calls for some channels or tasks and UI execution for others [1].	Legitimate if disclosed per channel. Dangerous if the mix is hidden [1].
Parametric memory	An answer generated from model weights without live web retrieval [1].	Can produce mentions and citations that no real user would see [1].
Retrieval	The product's own decision to search the web and select sources [1].	Small upstream ranking changes can change which brands get cited [1].
Rendered answer	The final answer as displayed to a user, including source panels and modules [1].	This is the only surface that matches what buyers see [1].
Source chip	A citation element shown in the interface, such as a linked source or card [1].	A tool that never renders the interface cannot measure it [1].
Mention	A detected brand name or alias in an answer [1].	Detection logic, such as exact string, alias list, or fuzzy match, changes counts [1].
Citation	A source URL or reference attached to an answer [1].	Citation data can be missing or synthetic in API-only collection [1].
Visibility score	An aggregate number derived from mentions or citations over runs [1].	Meaningless without a denominator, prompt set, and time window [1].
Run	One prompt execution on one channel at one time.	Single runs are noise in a non-deterministic system [1].
Denominator	The number of runs, prompts, and periods behind a metric [1].	A score of 74 means nothing without it [1].
Sampling frequency	How often each prompt is executed per day or week [1].	One run per week is a poll, not a measurement [1].
Non-determinism	The property that identical prompts can produce different answers [1].	Any metric must report rates over runs, not single observations [1].
Model drift	Silent provider updates that change answers across brands [1].	A Tuesday jump may be a model change, not your content win [1].
Method change annotation	A visible note when collection method changes [1].	Without it, historical charts are silently rewritten [1].
Raw answer access	Click-through from a metric to the stored AI answers [1].	This is the difference between trust us and check us [1].

4. Decision Rules by Use Case

Not every use case needs the same fidelity. I use the following rules to avoid overpaying for scale or underpaying for truth. The rules assume you know your use case. If you do not, start with brand visibility tracking, because

that is the use case most likely to be misrepresented. API collection is not bad. It is simply a different instrument. The error is using an API instrument to answer a UI question. The source article makes this distinction [1].

Use case	Preferred collection	Minimum sampling	Audit requirement
Large-scale prompt research	API, disclosed	High volume, single runs may be acceptable for exploration	Method disclosure and model version [1]
Fanout or query expansion analysis	API, disclosed	High volume	Method disclosure [1]
Model comparison	API or UI, depending on question	Multiple runs per model	Model version and run count [1]
Brand visibility tracking	UI execution	Multiple runs per prompt per period [1]	Raw answer access [1]
Citation and source tracking	UI execution with retrieval	Multiple runs per prompt per period [1]	Raw source capture [1]
Interface module tracking	UI execution only	Multiple runs per prompt per period	Rendered answer capture [1]
Executive reporting	UI execution or disclosed hybrid	Daily execution with multiple runs [1]	Denominators, change log, raw answer access [1]
Competitive benchmark	UI execution	Multiple runs per prompt per period [1]	Exportable raw data [1]

5. Sampling and Metric Integrity Tests

Variance is not a bug in AI answers; it is a property of the system. The source article notes that the same prompt can produce different answers and that sampling is the methodological answer [1]. I compress vendor claims into four tests.

1. Denominator test: Ask how many runs, prompts, and periods produced the metric. If the vendor cannot answer, stop. A score without a denominator is decoration [1].
2. Repeat test: Run one prompt five times in the real product. Count how often your brand appears. If the count changes, the vendor must report rates over runs, not single observations [1].
3. Trace test: Pick any metric and ask to click through to the raw answers. If you cannot see the underlying text and sources, you cannot audit mention detection or sampling [1].
4. Drift test: Ask how model updates are flagged. If a provider update shifts results across every brand, the tool should annotate the chart, not present it as a ranking change [1].

I also ask for the seven methodology questions in writing [1]. The exact wording matters less than the written answer. A vendor with real sampling will give concrete numbers. A vendor without it will give adjectives [1].

Sampling math is simple. If a prompt has a 20 percent chance of mentioning your brand, one run tells you almost nothing. Ten runs give you a rate with visible uncertainty. The exact number of runs depends on how close your brand is to the model's decision boundary, but the principle is fixed: report rates, not single observations [1]. I also ask whether the tool runs prompts daily. The source article treats daily execution with multiple runs per prompt as the practical minimum for stable trend data [1]. Weekly single runs can only support rough directional reads [1].

6. Vendor Audit Workflow

I run this audit before I trust a dashboard. It takes one afternoon if you are organized.

1. Choose five commercially relevant prompts in your market's language [1].
2. Run each prompt five times in the real consumer products: ChatGPT, Perplexity, Google with AI Overviews and AI Mode, and Gemini. Use your target country or a clean profile [1].
3. Record every brand and every source cited, run by run. Expect variation. That variation is your baseline [1].
4. Compare your manual baseline with the vendor's trial dashboard for those same five queries. Check cited URLs, brand mention rank, and citation chips [1].
5. Ask for written answers to the methodology questions. Request per-channel collection method, model version, run frequency, mention detection logic, geography and language controls, change announcements, and raw data export [1].
6. Request a reproduction test: can you export raw data and recompute the aggregate yourself? If not, mark traceability as failed [1].
7. Score the vendor on four dimensions: method disclosure, sampling adequacy, raw answer access, and change history. A vendor that fails raw answer access is not a benchmark candidate [1].

I score each dimension on a simple scale: documented, partially documented, or absent. I do not need a vendor to be perfect on every dimension, but I do need to know which dimensions are weak. A tool with excellent sampling but no raw answer access is still unauditable. A tool with raw answer access but no per-channel method disclosure is still ambiguous. The audit is not about finding a perfect vendor. It is about knowing where the data can mislead you.

7. Worked Examples

These examples show how I apply the rules. They are composites, not case studies. The examples are not vendor reviews. They are decision drills. In each case, the manual baseline is the control. The vendor dashboard is the treatment. If the treatment diverges from the control, I do not assume bad faith. I assume a methodological difference, then ask the vendor to explain it. If they cannot explain it, the difference becomes a risk. If they can explain it, I update my decision rule.

Scenario	What I observe	Test result	Decision rule
API-only vendor with high prompt volume and low price	No methodology page; dashboard shows 100% visibility for a brand [1].	My manual runs omit the brand 4 out of 5 times [1].	Use for exploratory research if disclosed, never for buyer-facing visibility reporting [1].
UI vendor with lower prompt volume and raw answers	Method page lists per-channel UI execution, model versions, and run counts [1].	Manual baseline matches dashboard citations and mention ranks [1].	Benchmark candidate. Proceed to sampling and drift checks [1].
Hybrid vendor with mixed channels	API for some channels, UI for others, but no per-channel labels [1].	Manual baseline matches on UI channels, diverges on API channels [1].	Conditional. Require per-channel labels, model versions, and change log before use [1].
Vendor with a two-digit monthly price for millions of prompts across ten channels [1]	No raw answer access; metrics have no denominators [1].	Cannot trace any metric to an observation [1].	No-go for audit. The scale is likely API or cache based, which may be fine, but the disclosure is missing [1].

8. Go/No-Go Checklist

I use this checklist as a hard gate. A single no-go on raw answer access is enough to reject a tool for benchmark work. The checklist is intentionally binary. I do not average scores across dimensions, because a missing raw answer access cannot be compensated by a nice UI. For benchmark work, raw answer access is a hard requirement. For exploratory research, I relax that requirement but still demand method disclosure. The decision rule depends on the cost of being wrong. If the metric will inform budget or board reporting, use the strict checklist. If it will inform a hypothesis, use the relaxed checklist.

- Go if the vendor documents collection method per channel in writing [1].
- Go if the vendor names the exact model or product version per channel [1].
- Go if the vendor reports runs per prompt per period and computes rates over runs [1].
- Go if the vendor explains mention detection logic and lets you inspect raw answers [1].
- Go if the vendor controls or discloses geography and language [1].
- Go if the vendor announces method changes and annotates historical data [1].
- Go if the vendor allows raw data export and reproduction of aggregates [1].
- No-go if the vendor guarantees outcomes, such as getting you into ChatGPT answers [1].
- No-go if the vendor shows metrics without denominators [1].
- No-go if the vendor silently rewrites history [1].
- No-go if the vendor cannot show the individual AI answers behind a score [1].
- No-go if two or more red flags appear together [1].

REFERENCES & SOURCES

[1] Source article: How AI Visibility Tools Actually Collect Data, https://metehan.ai/articles/how-ai-visibility-tools-collect-data/