LLMS.TXT · TECHNICAL SEO · AI CRAWLERS · IMPLEMENTATION

The llms.txt field guide: what it is, how to implement it, and does it actually work

llms.txt does not improve AI citation rates. AI assistant crawlers largely don't fetch it. Its one documented use case is developer tooling and MCP integrations. Here's the implementation guide.

AIVIARA RESEARCH · MAY 2026

Here is the verdict before anything else: llms.txt does not improve your AI citation rates. For B2B brands hoping it will help them appear in vendor-comparison queries or buyer-research answers, two large-scale citation studies found no meaningful effect, and crawler log analysis helps explain why. AI assistant crawlers are not consistently fetching the file at meaningful scale. Cloudflare's April 2026 Agent Readiness framework does not treat it as a default signal.

That does not mean you should ignore it. For one specific use case, it has real, documented utility. And it costs an afternoon to implement.

Read the evidence. Make a decision. Don't let the hype make it for you.

What llms.txt is

Jeremy Howard, co-founder of Answer.AI, proposed the format on September 3, 2024. It is a plain-text markdown file, placed at the root of your domain, that gives AI systems a curated guide to your most important content.

No W3C or IETF backing. An informal community proposal, managed through a GitHub repository and Discord channel. Howard described it as "open for community input." That is still where it stands, twenty months later.

The rationale behind the proposal was reasonable. HTML pages are noisy. Context windows have limits. Site owners understand their own content better than any crawler does. Let them curate a short list of what matters most, in a format AI can read cleanly.

The companion proposal is `/llms-full.txt`, a complete content export in Markdown rather than a curated index. More on this later, because the crawler data on full exports is more interesting than on the index file.

The format

One element is required: an H1 heading with your site or project name. Everything else is optional.

The full spec template:

```
# Project Name

> Brief description of what this site is and who it serves

Optional prose explaining the project in more detail.

## Section Name

- [Page title](https://example.com/page): One factual sentence about what this page covers.
- [Page title](https://example.com/page): One factual sentence about what this page covers.

## Another Section

- [Page title](https://example.com/page): One factual sentence about what this page covers.

## Optional

- [Less critical page](https://example.com/secondary): Lower-priority reference material.
```

The H2 sections organise links by content area. The "Optional" H2 at the end signals lower-priority content that a system working under context constraints can skip.

Link entry format: `Page title: one factual sentence about what this page covers.`

That last part matters more than most implementations get right.

Some examples of this done well. Stripe organises its file by product area ("Payments API: Charges and Payment Intents"). Supabase publishes separate language-segmented exports for JavaScript, Python, and Dart rather than one monolithic file. Cloudflare covers more than 20 products with substantial per-section depth. Anthropic runs its own file at docs.anthropic.com/llms.txt. These are deliberate, maintained reference documents.

What AI engines actually do with it

John Mueller, Google Search liaison, on Bluesky, June 17, 2025: "FWIW no AI system currently uses llms.txt."

Mueller elaborated. AI assistants and chatbots fetch your pages for training and grounding, but none of them fetch the llms.txt file. You can verify this in your server logs.

In April 2025, on Reddit, Mueller compared llms.txt to the keywords meta tag. "AFAIK none of the AI services have said they're using LLMs.TXT (and you can tell when you look at your server logs that they don't even check for it)." Reported by Roger Montti at Search Engine Journal.

On December 3, 2025, Google Search Central briefly published an llms.txt file at developers.google.com/search/docs/llms.txt. It was removed the same day. Mueller's response when asked about it was "hmmn :-/". No official reversal of his stated position has followed.

The server log data backs him up. Flavio Longato at Adobe ran a 30-day CDN log analysis across 1,000 domains in August 2025. Result: zero GPTBot, ClaudeBot, or PerplexityBot requests to llms.txt files. The bot making 94.9% of requests to those files was GoogleBotDesktop. OpenAIBotSearch accounted for 1.1%.

> In a 30-day CDN log analysis across 1,000 domains: zero GPTBot, ClaudeBot, or PerplexityBot requests to llms.txt files. 94.9% of requests came from GoogleBotDesktop.
>
> FLAVIO LONGATO, ADOBE CDN ANALYSIS — AUGUST 2025

Perplexity has its own llms.txt (4K tokens) and llms-full.txt (177K tokens). Longato found zero PerplexityBot requests to third-party llms.txt files in his sample.

Anthropic has the strongest documented engagement signal. Anthropic operates its own file at docs.anthropic.com/llms.txt and specifically asked Mintlify to implement llms.txt for documentation. ClaudeBot crawling of the file has been documented. Anthropic has not published official documentation explaining what ClaudeBot does with it in citation selection.

Mintlify and Profound ran a 7-day CDN log study across 25 companies, led by Tiffany Chen. Median visits to llms.txt files: 14. Median visits to llms-full.txt: 79. ChatGPT accounted for the majority of llms-full.txt traffic.

So some crawling is happening, particularly to the full export. The question is whether any of it affects what gets cited in answers. The citation rate evidence says no.

The one use case with documented, real-world utility is developer tooling. LangChain built the mcpdoc MCP server, which reads llms.txt files to expose documentation structure to IDE-integrated agents like Cursor and Claude Code. Mintlify now auto-generates both llms.txt and MCP servers from the same source. For products with developer audiences and documentation-heavy content, this is a real workflow, not a theoretical one.

The citation rate evidence

Two large-scale studies. Both negative.

SE Ranking, led by Yulia Deda, November 7, 2025: approximately 300,000 domains, multiple methods including Spearman correlation analysis, XGBoost regression, and SHAP analysis. Finding: no relationship between having llms.txt and citation rates in AI answers. When the team removed llms.txt from the model variables, accuracy improved. The variable was adding noise, not signal. Adoption rate in the sample: 10.13% of domains had the file.

> SE Ranking's analysis of 300,000 domains found no relationship between llms.txt adoption and AI citation rates. Removing the variable from their model improved prediction accuracy.
>
> SE RANKING — 300,000 DOMAINS, NOVEMBER 2025

Generix Marketing, led by Jon Eric Dela Cruz, April 15, 2026: 2,500 top sites by organic traffic, 156 confirmed llms.txt files (6.5% adoption), 656 prompts run across Perplexity, ChatGPT, and Claude via API, 2,128 total prompt/citation checks. Sites with llms.txt captured 8% of citations against a 6.5% baseline expectation, a 1.27x over-representation. Dela Cruz's own assessment was that this does not clear the statistical threshold for significance, and confounders including domain authority and content depth were not controlled.

The mechanistic explanation for why both studies find no effect is straightforward. If AI assistant crawlers are not fetching the file, it cannot influence citation behaviour through real-time retrieval. The Longato log analysis provides that explanation.

What actually moves AI citation rates, according to Vlad Kuriatnyk at The Digital Bloom (December 2025, Princeton GEO analysis methodology, 10,000 queries, 680 million citations):

Brand search volume is the strongest predictor, at a 0.334 correlation coefficient. Statistics in content add 22% visibility. Quotations from named sources add 37%. Cross-platform presence on four or more platforms makes a brand 2.8 times more likely to appear. Content

recency: 65% of AI crawler hits target content from the past year.

llms.txt does not appear on that list.

Six mistakes worth avoiding

Dumping your sitemap into the file. The whole point of llms.txt is curation. A 400-link file that mirrors your sitemap tells a model nothing useful. Pick your 5 to 30 most important pages. If everything is a priority, nothing is.

Marketing language in descriptions. This is the most common mistake. Two versions of the same link entry:

Bad: `Enterprise Security: We provide industry-leading security solutions to protect your most critical assets.`

Good: `Enterprise Security: Covers SOC 2 Type II compliance, encryption standards, and access control configuration for enterprise deployments.`

The first example is generic marketing copy. The second actually tells a model what is on the page. AI models are parsing these descriptions for content signals, not reading them as a prospective customer would.

Wrong placement. Root only. `yourdomain.com/llms.txt`. Not `/docs/llms.txt`. Not `/resources/llms.txt`. A subdirectory file will not be found by crawlers looking for the standard location.

Incorrect Content-Type header. The server must return `text/plain`. Other content types break parsing. Check this after deployment, not before.

Confusing it with robots.txt. robots.txt is an exclusion mechanism. It tells crawlers what they cannot access. llms.txt is an inclusion guide. It tells them what is most worth reading. Crawl rules do not belong in llms.txt. This confusion shows up in real files.

Letting it go stale. An outdated file with discontinued products and dead links may be worse than no file at all. If a model treats your description as accurate, wrong information causes active harm. If you cannot commit to updating it on major product changes, the case for having it weakens considerably.

How to build and deploy it

Step 1: Audit your content. Identify your 5 to 30 most important pages. For B2B SaaS this typically means core product and feature pages, key use case pages, case studies, security documentation, API reference, and FAQ or glossary content. Exclude parameter variants, tag pages, paginated pages, press releases, and staging URLs.

Step 2: Organise by intent, not navigation. Your site nav is structured around how your marketing team thinks about your product. Your llms.txt should be structured around how a model would use the content. Sections like "Getting Started," "Core Product," "Use Cases," "API Reference," and "Case Studies" serve that purpose better than your navigation labels do.

Step 3: Write factual descriptions. Use the format: `Page title: one factual sentence about what this page covers.` Concrete noun phrases. Technical terms where relevant. No superlatives, no marketing language. If you cannot describe a page in one factual sentence, that is useful information about whether the page is well-scoped.

A finished B2B SaaS implementation looks something like this:

```
# Acme Analytics

> Analytics and attribution platform for B2B SaaS companies tracking pipeline from marketing.

## Core product

- [How Acme works](https://acmeanalytics.com/how-it-works): Explains the attribution model, data
ingestion pipeline, and supported CRM integrations.
- [Pricing](https://acmeanalytics.com/pricing): Covers per-seat and usage-based plans; enterprise
pricing requires a sales call.

## Use cases

- [Marketing attribution](https://acmeanalytics.com/use-cases/marketing): How marketing teams use
Acme to connect campaigns to pipeline.
- [Revenue ops](https://acmeanalytics.com/use-cases/revops): How RevOps teams use Acme for
multi-touch attribution across the funnel.

## Optional

- [Blog](https://acmeanalytics.com/blog): Articles on B2B attribution methodology.
- [Changelog](https://acmeanalytics.com/changelog): Product updates and release notes.
```

Concise, factual, and organised around how a model would use the content rather than how the nav is laid out.

Step 4: Decide between index and full export. An llms.txt index file links to your important pages. An llms-full.txt exports the complete content in Markdown. The Mintlify/Profound crawler data shows llms-full.txt gets 3 to 4 times more visits than the index. Both can coexist. For sites where the full content export is tractable, publishing both is reasonable.

Step 5: Deploy at root. Verify. The file must be accessible at `https://yourdomain.com/llms.txt`. Confirm HTTP 200 response. Confirm `Content-Type: text/plain`. No DNS record, meta tag, or server configuration required beyond those two things. If you are behind a CDN, invalidate the cache after any update.

Step 6: Test and set a schedule. Fetch the file in a browser. Validate the markdown structure. Add a `Last-updated: YYYY-MM` line near the top. Set a calendar reminder to update on major product changes. Quarterly review at minimum.

Where it fits

A quick map, because these get conflated:

robots.txt. Crawler access control. Tells bots what they cannot access. Exclusion mechanism.

sitemap.xml. Complete page catalogue. Every indexable URL on your site. No curation, no description. Exhaustive.

llms.txt. Curated reading list. A selective guide to what matters most, in a format AI can parse cleanly. Inclusion and description mechanism.

Structured data (schema.org). Marks up content within individual pages. FAQ schema, Article schema, HowTo schema, Product schema. The citation rate evidence for structured data is stronger than for llms.txt. Kurt Fischman's controlled test found pages with JSON-LD FAQ schema achieved 41% citation rates against 24% for equivalent pages without it. Structured data and llms.txt operate at different layers and are not substitutes.

MCP (Model Context Protocol). Anthropic released MCP in November 2024 as an open protocol for agentic workflows. It is a different layer entirely, enabling AI models to connect to external tools, data sources, and APIs. LangChain's mcpdoc server reads llms.txt to expose documentation structure to IDE-integrated agents. Mintlify generates llms.txt and MCP servers from the same source. In practice, llms.txt acts as a lightweight documentation index for MCP-connected tooling. MCP then provides the protocol layer that lets agents query tools and systems directly.

Cloudflare published its Agent Readiness Score framework in April 2026 (André Jesus and Vance Morrison). Four evaluation dimensions: discoverability (robots.txt, sitemap.xml, Link Headers), content (Markdown support), bot access control (AI bot rules, Web Bot Auth), and capabilities (MCP Server Card, API Catalog, OAuth discovery). llms.txt is listed as an optional

add-on, not a default signal.

What this won't do

It will not improve your inclusion in AI assistant answers — ChatGPT, Perplexity, Google AI Overviews, Claude.ai. Two large-scale studies find no effect. The mechanistic reason is straightforward: if AI assistant crawlers are not consistently fetching the file, it cannot influence what gets cited.

It will not correct AI misinformation about your brand. If models are describing your products inaccurately — wrong pricing, discontinued features, attributes borrowed from a competitor — that is a training data and third-party source problem. llms.txt does not reach training data, and it does not override what AI systems have already learned from other sources.

It will not provide citation tracking or analytics. You will not know whether any AI system read your file or acted on it. That visibility does not exist.

It will not protect you from crawlers ignoring it. There is no mechanism to enforce compliance with llms.txt content. Any AI system can crawl anything it has permission to access, regardless of what your file says.

For AI answer visibility, the signals that actually matter are brand search volume, statistics and quotations in your content, cross-platform presence, and content recency. Those are where the evidence points. llms.txt is not in that list.

llms.txt costs an afternoon to implement and requires quarterly maintenance. For a B2B SaaS company with a developer audience, or a documentation-heavy product where IDE integrations matter, the MCP and developer tooling use case is real and documented.

For AI answer inclusion in 2026, no evidence it helps. If a major AI company officially adopts it as a citation signal, that could change. None has, in twenty months.

Track it. Don't bet on it.

Aiviara is building infrastructure for monitoring AI brand citations and factual accuracy across LLM platforms. Early access information is available at aiviara.com.