PRACTITIONER RESEARCH WHITE PAPER · NOVEMBER 2026

Do AI Crawlers Follow /llms.txt and Markdown Twins?

Session Journey Forensics Across 225,276 Origin Server Log Requests
Why Standard .md Twin and PDF Fetches Exceed 23,000 Direct Hits

Tygart Media · tygartmedia.com · Anonymized Multi-Property Telemetry Edition

30	96.43%	10,807	12,455
Total /llms.txt Hits	**Zero-.md Follow-Through**	**Markdown Twin Hits**	**PDF Asset Hits**
0 from major AI labs	27 of 28 sessions abandon	via HTML or cold cache	Consistent legacy crawler pull

Executive Summary

When a new standard for AI readability emerges, the tech and SEO communities frequently adopt it before verifying if major crawl fleets actually utilize it. The `/llms.txt` convention proposed a central manifest to help AI agents discover machine-readable Markdown documentation. To test adoption in late 2026, we audited **225,276** non-operator origin HTTP requests across eight B2B publishing properties (Sites 01–08), performing chronological `(site, ip, ua)` journey forensics on how bots locate and consume content.

The findings show a massive disconnect between developer enthusiasm and actual crawler behavior. The `/llms.txt` file received just **30 hits** in total — with **zero** hits from GPTBot, ClaudeBot, PerplexityBot, or Googlebot. Worse, when crawlers did fetch `/llms.txt`, 96.43% of sessions abandoned the journey without ever fetching a `.md` Markdown twin. In contrast, AI crawlers and traditional indexers pulled **10,807 Markdown twins** and **12,455 PDF files** directly, discovering them via standard HTML link headers and direct cached indexation rather than a central `/llms.txt` manifest.

1. The /llms.txt Paradox vs. Direct Machine Consumption

The theory behind `/llms.txt` is elegant: provide a `robots.txt`-like map of Markdown equivalents so AI agents do not have to parse messy HTML DOM trees. However, AI labs already built their infrastructures around crawling the open web as it exists. OpenAI, Anthropic, and Perplexity indexers are perfectly capable of extracting `<link rel="alternate" type="text/markdown">` tags from standard HTML headers, or simply discovering URLs via XML sitemaps.

As a result, building `/llms.txt` currently caters almost exclusively to niche hobbyist bots or localized scripts, rather than the major AI engines responsible for citation generation. If you build Markdown `.md` twins for your articles and documentation, AI bots will absolutely consume them — just not via `/llms.txt`.

KEY TAKEAWAY FOR CONTENT ENGINEERING

Stop waiting for AI bots to read `/llms.txt`. They are already consuming Markdown twins and PDFs at scale by discovering them directly in your HTML `<head>`. A decentralized alternate-link strategy works today; centralized manifests are being ignored by the major labs.

2. Chronological (site, ip, ua) Journey Forensics Methodology

A raw hit count does not tell the full story of discovery. To understand *how* bots found Markdown twins, we performed session reconstruction. We grouped the 225,276 requests by the combination of `Site ID`, `IP Address`, and `User-Agent`, applying a 30-minute idle timeout boundary to separate distinct crawl passes. By analyzing the chronological sequence of requests within these 12,432 twin-or-pdf sessions, we mapped the discovery paths AI crawlers actually use.

Across 28 sessions that began by fetching `/llms.txt`, only **1 single session** followed up by fetching a Markdown `.md` file. The remaining 27 sessions (96.43%) were dead ends: 16 sessions fetched `/llms.txt` and stopped crawling entirely, while 11 sessions abandoned the manifest and fell back to scraping standard HTML pages instead.

If they aren't using `/llms.txt`, how did bots generate 10,807 Markdown hits and 12,455 PDF hits?

- **HTML-First Discovery:** 1,103 Markdown sessions and 1,823 PDF sessions began by fetching standard HTML pages first. The bot parses the HTML, finds the alternate link or direct URL, and immediately fetches the machine-readable version.
- **Cold-Direct Cached Fetching:** 2,626 Markdown sessions and 3,829 PDF sessions were "cold" direct hits. The bot arrived at the `.md` or `.pdf` URL directly from its own internal cache or index, without warming up on HTML or `/llms.txt` during that session.

FIGURE 1: CHRONOLOGICAL SESSION FORENSICS (3 PATHS TO MACHINE READABILITY)



The Fallacy of the Manifest

These forensic paths reveal that major crawlers process content asynchronously. They discover URLs through sitemaps or HTML crawls, enqueue those URLs in massive distributed caching systems, and fetch them directly hours or days later (Path C). A monolithic `/llms.txt` file introduces a synchronous bottleneck that modern distributed web crawlers actively avoid.

3. 15-Family /llms.txt vs. .md vs. .pdf Consumption Matrix

The table below breaks down the 10,807 Markdown hits and 12,455 PDF hits across CIDR/ASN-verified bot families. Note the heavy reliance on `.md` twins by major AI labs and the absolute zero volume on `/llms.txt` from top-tier models:

Verified Bot Family	/llms.txt	.md Twin	.md 200	.md 304	.pdf Total
GPTBot	0	397	375	0 (22 404)	394
ClaudeBot	0	388	386	2	453
PerplexityBot	0	313	303	10	260
OAI-SearchBot	0	345	205	140	142
ChatGPT-User	0	0	0	0	55
Claude-User	0	7	7	0	1
Claude-Code-CLI	0	11	11	0	0
Googlebot	0	463	305	158	302
Bingbot	0	489	207	156 (126 404)	992
ExaSearchBot	0	120	84	36	146
Meta-ExternalAgent	0	450	—	—	722
Bytespider	0	624	—	—	704
Amazonbot	1	415	—	—	475
Likely-Human-Browser	14	3,043	—	—	3,828
Other Compatible / Crawlers	13	2,124	—	—	1,869
Grand Total	30	10,807	8,236	2,388 (178 404)	12,455

4. Pre-Deploy 404 Race Conditions and Alternate Link Discovery

Because indexers cache URLs heavily, they often re-crawl Markdown twins based on past discovery. This creates a race condition for publishers: if an AI crawler discovers a `.md` alternate link in your HTML, caches it, and attempts to re-fetch it months later after you have migrated or deleted the `.md` file, the bot will hit a `404 Not Found`.

In our dataset, we recorded 178 `.md` `404` errors across the 8-site corpus (led by Bingbot with 126 and GPTBot with 22). In many of these cases, we observed "pre-deploy" `404` race conditions where an HTML page was published slightly before its Markdown twin finished building. An aggressive indexer hitting the HTML immediately extracted the alternate link and fetched the `.md` URL before the file existed, causing a `404` that later resolved to a `200 OK` on subsequent passes.

This confirms beyond doubt that AI crawlers aggressively parse HTML headers for alternate links and immediately add those targets to their fetch queues, circumventing the need for a central manifest file entirely.

5. The 5-Step Machine-Readable Twin Playbook

If `/llms.txt` isn't moving the needle for AI citation engines, how should content and engineering teams build for machine readability? The data supports a localized, decentralized approach:

1. **Inject HTML Head Alternate Links:** Use standard alternate links in the `<head>` of your HTML pointing to your Markdown file. Crawlers already look for them.
2. **Maintain Clean Markdown Endpoints:** Serve real, sanitized Markdown without wrapper HTML or complex navigation menus. `Claude-Code-CLI` and `Claude-User` fetch `.md` endpoints directly to save token context.
3. **Use Deterministic HTML-to-MD Parity:** Ensure your Markdown twins are generated automatically from the same headless CMS or WordPress REST pipeline as your HTML. Manual drift destroys citation trust.
4. **Do Not Ignore PDFs:** Despite being a legacy format, PDFs drew 12,455 hits across our dataset (outpacing `.md`). AI engines maintain robust text extraction pipelines. Keep them accessible and text-selectable.
5. **Monitor Your Own Logs, Not Hype:** Configure nginx/Apache logging to track `.md` and `.pdf` extensions against verified vendor ASNs. Ignore raw User-Agent greps and rely on CIDR validation to measure real AI traffic.

Primary Verification Sources

- **The /llms.txt Proposal:** llmstxt.org — The community proposal for providing Markdown to LLMs.
- **RFC 9309 (robots.txt):** rfc-editor.org/rfc/rfc9309.html — Standard exclusion protocol vs decentralized manifests.
- **AI Lab Crawler JSONs:** openai.com/searchbot.json, claude.com/crawling/bots.json, perplexity.ai/perplexitybot.json.
- **Part 1 Origin Log Study:** tygartmedia.com/do-ai-crawlers-render-javascript-server-log-white-paper/ — Analyzing JS rendering across the same 225,276-request dataset.

ABOUT THIS RESEARCH

Published by Tygart Media (tygartmedia.com) on November 23, 2026. All traffic counts reflect anonymized server access logs across eight independent WordPress properties audited with strict vendor IP/ASN verification. Client identities, internal hostnames, and exact IPs have been omitted.