The text file that runs the internet

For three decades, a tiny text file has kept the internet from chaos. This text file has no particular legal or technical authority, and it's not even particularly complicated. It represents a handshake deal between some of the earliest pioneers of the internet to respect each other's wishes and build the internet in a way that benefitted everybody. It's a mini constitution for the internet, written in code.

It's called robots.txt and is usually located at yourwebsite.com/robots.txt. That file allows anyone who runs a website — big or small, cooking blog or multinational corporation — to tell the web who's allowed in and who isn't. Which search engines can index your site? What archival projects can grab a version of your page and save it? Can competitors keep tabs on your pages for their own files? You get to decide and declare that to the web.

It's not a perfect system, but it works. Used to, anyway. For decades, the main focus of robots.txt was on search engines; you'd let them scrape your site and in exchange they'd promise to send people back to you. Now AI has changed the equation: companies around the web are using your site and its data to build massive sets of training data, in order to build models and products that may not acknowledge your existence at all.

The robots.txt file governs a give and take; AI feels to many like all take and no give. But there's now so much money in AI, and the technological state of the art is changing so fast that many site owners can't keep up. And the fundamental agreement behind robots.txt, and the web as a whole — which for so long amounted to "everybody just be cool" — may not be

able to keep up either.

In the early days of the internet, robots went by many names: spiders, crawlers, worms, WebAnts, web crawlers. Most of the time, they were built with good intentions. Usually it was a developer trying to build a directory of cool new websites, make sure their own site was working properly, or build a research database – this was 1993 or so, long before search engines were everywhere and in the days when you could fit most of the internet on your computer's hard drive.

The only real problem then was the traffic: accessing the internet was slow and expensive both for the person seeing a website and the one hosting it. If you hosted your website on your computer, as many people did, or on hastily constructed server software run through your home internet connection, all it took was a few robots overzealously downloading your pages for things to break and the phone bill to spike.

Over the course of a few months in 1994, a software engineer and developer named Martijn Koster, along with a group of other web administrators and developers, came up with a solution they called the Robots Exclusion Protocol. The proposal was straightforward enough: it asked web developers to add a plain-text file to their domain specifying which robots were not allowed to scour their site, or listing pages that are off limits to all robots. (Again, this was a time when you could maintain a list of every single robot in existence – Koster and a few others helpfully did just that.) For robot makers, the deal was even simpler: respect the wishes of the text file.

From the beginning, Koster made clear that he didn't hate robots, nor did he intend to get rid of them. "Robots are one of the few aspects of the web that cause operational problems and cause people grief," he said in an initial email to a mailing list called WWW-Talk (which included early-internet pioneers like Tim Berners-Lee and Marc Andreessen) in early

1994. “At the same time they do provide useful services.” Koster cautioned against arguing about whether robots are good or bad — because it doesn’t matter, they’re here and not going away. He was simply trying to design a system that might “minimise the problems and may well maximize the benefits.”

“Robots are one of the few aspects of the web that cause operational problems and cause people grief. At the same time, they do provide useful services.”

By the summer of that year, his proposal had become a standard — not an official one, but more or less a universally accepted one. Koster pinged the WWW-Talk group again in June with an update. “In short it is a method of guiding robots away from certain areas in a Web server’s URL space, by providing a simple text file on the server,” he wrote. “This is especially handy if you have large archives, CGI scripts with massive URL subtrees, temporary information, or you simply don’t want to serve robots.” He’d set up a topic-specific mailing list, where its members had agreed on some basic syntax and structure for those text files, changed the file’s name from RobotsNotWanted.txt to a simple robots.txt, and pretty much all agreed to support it.

And for most of the next 30 years, that worked pretty well.

But the internet doesn’t fit on a hard drive anymore, and the robots are vastly more powerful. Google uses them to crawl and index the entire web for its search engine, which has become the interface to the web and brings the company billions of dollars a year. Bing’s crawlers do the same, and Microsoft licenses its database to other search engines and companies. The Internet Archive uses a crawler to store webpages for posterity. Amazon’s crawlers traipse the web looking for product information, and according to a recent antitrust suit, the company uses that information to punish sellers who offer better deals away from Amazon. AI companies like OpenAI are crawling the web in order to train large language models that

could once again fundamentally change the way we access and share information.

The ability to download, store, organize, and query the modern internet gives any company or developer something like the world's accumulated knowledge to work with. In the last year or so, the rise of AI products like ChatGPT, and the large language models underlying them, have made high-quality training data one of the internet's most valuable commodities. That has caused internet providers of all sorts to reconsider the value of the data on their servers, and rethink who gets access to what. Being too permissive can bleed your website of all its value; being too restrictive can make you invisible. And you have to keep making that choice with new companies, new partners, and new stakes all the time.

There are a few breeds of internet robot. You might build a totally innocent one to crawl around and make sure all your on-page links still lead to other live pages; you might send a much sketchier one around the web harvesting every email address or phone number you can find. But the most common one, and the most currently controversial, is a simple web crawler. Its job is to find, and download, as much of the internet as it possibly can.

Web crawlers are generally fairly simple. They start on a well-known website, like cnn.com or wikipedia.org or health.gov. (If you're running a general search engine, you'll start with lots of high-quality domains across various subjects; if all you care about is sports or cars, you'll just start with car sites.) The crawler downloads that first page and stores it somewhere, then automatically clicks on every link on that page, downloads all those, clicks all the links on every one, and spreads around the web that way. With enough time and enough computing resources, a crawler will eventually find and download billions of webpages.

The tradeoff is fairly straightforward: if Google can crawl

your page, it can index it and show it in search results.

Google estimated in 2019 that more than 500 million websites had a robots.txt page dictating whether and what these crawlers are allowed to access. The structure of those pages is usually roughly the same: it names a “User-agent,” which refers to the name a crawler uses when it identifies itself to a server. Google’s agent is Googlebot; Amazon’s is Amazonbot; Bing’s is Bingbot; OpenAI’s is GPTBot. Pinterest, LinkedIn, Twitter, and many other sites and services have bots of their own, not all of which get mentioned on every page. (Wikipedia and Facebook are two platforms with particularly thorough robot accounting.) Underneath, the robots.txt page lists sections or pages of the site that a given agent is not allowed to access, along with specific exceptions that are allowed. If the line just reads “Disallow: /” the crawler is not welcome at all.

It’s been a while since “overloaded servers” were a real concern for most people. “Nowadays, it’s usually less about the resources that are used on the website and more about personal preferences,” says John Mueller, a search advocate at Google. “What do you want to have crawled and indexed and whatnot?”

The biggest question most website owners historically had to answer was whether to allow Googlebot to crawl their site. The tradeoff is fairly straightforward: if Google can crawl your page, it can index it and show it in search results. Any page you want to be Googleable, Googlebot needs to see. (How and where Google actually displays that page in search results is of course a completely different story.) The question is whether you’re willing to let Google eat some of your bandwidth and download a copy of your site in exchange for the visibility that comes with search.

For most websites, this was an easy trade. “Google is our most important spider,” says Medium CEO Tony Stubblebine. Google

gets to download all of Medium's pages, "and in exchange we get a significant amount of traffic. It's win-win. Everyone thinks that." This is the bargain Google made with the internet as a whole, to funnel traffic to other websites while selling ads against the search results. And Google has, by all accounts, been a good citizen of robots.txt. "Pretty much all of the well-known search engines comply with it," Google's Mueller says. "They're happy to be able to crawl the web, but they don't want to annoy people with it… it just makes life easier for everyone."

In the last year or so, though, the rise of AI has upended that equation. For many publishers and platforms, having their data crawled for training data felt less like trading and more like stealing. "What we found pretty quickly with the AI companies," Stubblebine says, "is not only was it not an exchange of value, we're getting nothing in return. Literally zero." When Stubblebine announced last fall that Medium would be blocking AI crawlers, he wrote that "AI companies have leached value from writers in order to spam Internet readers."

Over the last year, a large chunk of the media industry has echoed Stubblebine's sentiment. "We do not believe the current 'scraping' of BBC data without our permission in order to train Gen AI models is in the public interest," BBC director of nations Rhodri Talfan Davies wrote last fall, announcing that the BBC would also be blocking OpenAI's crawler. *The New York Times* blocked GPTBot as well, months before launching a suit against OpenAI alleging that OpenAI's models "were built by copying and using millions of *The Times*'s copyrighted news articles, in-depth investigations, opinion pieces, reviews, how-to guides, and more." A study by Ben Welsh, the news applications editor at *Reuters*, found that 606 of 1,156 surveyed publishers had blocked GPTBot in their robots.txt file.

It's not just publishers, either. Amazon, Facebook, Pinterest,

WikiHow, WebMD, and many other platforms explicitly block GPTBot from accessing some or all of their websites. On most of these robots.txt pages, OpenAI's GPTBot is the only crawler explicitly and completely disallowed. But there are plenty of other AI-specific bots beginning to crawl the web, like Anthropic's anthropic-ai and Google's new Google-Extended. According to a study from last fall by Originality.AI, 306 of the top 1,000 sites on the web blocked GPTBot, but only 85 blocked Google-Extended and 28 blocked anthropic-ai.

There are also crawlers used for both web search and AI. CCBot, which is run by the organization Common Crawl, scours the web for search engine purposes, but its data is also used by OpenAI, Google, and others to train their models. Microsoft's Bingbot is both a search crawler and an AI crawler. And those are just the crawlers that identify themselves — many others attempt to operate in relative secrecy, making it hard to stop or even find them in a sea of other web traffic. For any sufficiently popular website, finding a sneaky crawler is needle-in-haystack stuff.

In large part, GPTBot has become the main villain of robots.txt because OpenAI allowed it to happen. The company published and promoted a page about how to block GPTBot and built its crawler to loudly identify itself every time it approaches a website. Of course, it did all of this *after* training the underlying models that have made it so powerful, and only once it became an important part of the tech ecosystem. But OpenAI's chief strategy officer Jason Kwon says that's sort of the point. "We are a player in an ecosystem," he says. "If you want to participate in this ecosystem in a way that is open, then this is the reciprocal trade that everybody's interested in." Without this trade, he says, the web begins to retract, to close — and that's bad for OpenAI and everyone. "We do all this so the web can stay open."

By default, the Robots Exclusion Protocol has always been permissive. It believes, as Koster did 30 years ago, that most

robots are good and are made by good people, and thus allows them by default. That was, by and large, the right call. “I think the internet is fundamentally a social creature,” OpenAI’s Kwon says, “and this handshake that has persisted over many decades seems to have worked.” OpenAI’s role in keeping that agreement, he says, includes keeping ChatGPT free to most users — thus delivering that value back — and respecting the rules of the robots.

But robots.txt is not a legal document — and 30 years after its creation, it still relies on the good will of all parties involved.

But robots.txt is not a legal document — and 30 years after its creation, it still relies on the good will of all parties involved. Disallowing a bot on your robots.txt page is like putting up a “No Girls Allowed” sign on your treehouse — it sends a message, but it’s not going to stand up in court. Any crawler that wants to ignore robots.txt can simply do so, with little fear of repercussions. (There is some legal precedent around web scraping in general, though even that can be complicated and mostly lands on crawling and scraping being allowed.) The Internet Archive, for example, simply announced in 2017 that it was no longer abiding by the rules of robots.txt. “Over time we have observed that the robots.txt files that are geared toward search engine crawlers do not necessarily serve our archival purposes,” Mark Graham, the director of the Internet Archive’s Wayback Machine, wrote at the time. And that was that.

As the AI companies continue to multiply, and their crawlers grow more unscrupulous, anyone wanting to sit out or wait out the AI takeover has to take on an endless game of whac-a-mole. They have to stop each robot and crawler individually, if that’s even possible, while also reckoning with the side effects. If AI is in fact the future of search, as Google and others have predicted, blocking AI crawlers could be a short-term win but a long-term disaster.

There are people on both sides who believe we need better, stronger, more rigid tools for managing crawlers. They argue that there's too much money at stake, and too many new and unregulated use cases, to rely on everyone just agreeing to do the right thing. "Though many actors have some rules self-governing their use of crawlers," two tech-focused attorneys wrote in a 2019 paper on the legality of web crawlers, "the rules as a whole are too weak, and holding them accountable is too difficult."

Some publishers would like more detailed controls over both what is crawled and what it's used for, instead of robots.txt's blanket yes-or-no permissions. Google, which a few years ago made an effort to make the Robots Exclusion Protocol an official formalized standard, has also pushed to deemphasize robots.txt on the grounds that it's an old standard and too many sites don't pay attention to it. "We recognize that existing web publisher controls were developed before new AI and research use cases," Google's VP of trust Danielle Romain wrote last year. "We believe it's time for the web and AI communities to explore additional machine-readable means for web publisher choice and control for emerging AI and research use cases."

Even as AI companies face regulatory and legal questions over how they build and train their models, those models continue to improve and new companies seem to start every day. Websites large and small are faced with a decision: submit to the AI revolution or stand their ground against it. For those that choose to opt out, their most powerful weapon is an agreement made three decades ago by some of the web's earliest and most optimistic true believers. They believed that the internet was a good place, filled with good people, who above all wanted the internet to be a good thing. In that world, and on that internet, explaining your wishes in a text file was governance enough. Now, as AI stands to reshape the culture and economy of the internet all over again, a humble plain-text file is

starting to look a little old-fashioned.

https://www.theverge.com/24067997/robots-txt-ai-text-file-web-crawlers-spiders