GitHub submission to IAB workshop on AI-CONTROL

GitHub welcomes the IAB's invitation to submit considerations regarding the suitability of the Robots Exclusion Protocol (RFC 9309) for communicating preferences regarding limits to AI training. The focus of this position paper is on the needs of software developers who wish to express such preferences.

General comments

Robots.txt is already widely used to address AI-related crawlers, though significant observed discrepancies between website owners' use of robots.txt and their stated preferences in terms and conditions point towards shortcomings of the protocol for this purpose. The same study has observed adverse effects of this emerging practice on the Internet ecosystem, including crawling for purposes of search or academic studies, which the workshop should consider before endorsing a REP-based practice of expressing preferences regarding AI crawling.

Distinguishing between service owners and copyright holders

There are at least two overlapping but distinct stakeholder groups that have an interest in expressing preferences regarding the crawling or scraping of publicly available web content as training data for AI models: service owners and copyright holders. **The workshop should distinguish between their needs**.

The Robots Exclusion Protocol (REP) is geared towards the needs of **service owners**[1]. It allows anyone with control over a domain to use robots.txt to request crawlers to respect crawling preferences. The practices of some AI-related crawlers have been reported to put an undue strain on service owners' resources. Those service owners may benefit from the ability to express preferences based on the purposes for which crawling takes place, rather than the identity of the crawlers. This goal could be achieved with or without changes to the REP, for example by establishing naming conventions for user agents. Such naming conventions could allow service owners to communicate preferences to categories of crawlers, such as all crawlers that collect data for the purposes of training AI models, regardless of the company or entity that employs the crawler. In any case, the success of such a model would depend on voluntary cooperation and agreement on the categorization of crawlers and corresponding generic user agents.

Several groups of **copyright holders** have expressed a strong interest in declaring their preferences regarding the use of their works for AI training. Such works are frequently collected

[1] "It may be inconvenient for service owners if crawlers visit the entirety of their URI space", RFC 9309, section 1.

through crawling or scraping of openly available web resources. Under certain circumstances, EU law may require AI developers to respect machine-readable opt-outs expressed by rights holders (Art. 4 Copyright in the Digital Single Market Directive (EU CDSMD) and Article 53 AI Act). Rights holders and AI developers have a shared interest in establishing an industry standard for machine-readable opt-outs to facilitate communication and establish a relationship of trust between the affected parties. Such a standard should include possibilities for rights holders to communicate different preferences regarding crawling for AI training purposes and crawling for other purposes, such as search.

While some rights holders may also be service owners, many are not. Many rights holders upload their works to online platforms, either directly or through third parties. Without direct control over the platform's domain, rights holders cannot use robots.txt to express opt-outs. As a consequence, some work has been done to communicate opt-outs independently of robots.txt, for example by the W3C TDMReP Community Group or Spawning.

Platform operators should be part of the conversation on communicating AI opt-outs, as they play an important role in communicating opt-out preferences of copyright holders to the operators of crawlers. It must be clear that any opt-out standard is solely a means of communicating AI opt-out preferences to third parties. Hosting platforms are not responsible for enforcing adherence to opt-outs by third parties, even if they choose to support their users in expressing opt-outs. It would be impractical for platform operators to express copyright holder opt-outs through robots.txt on behalf of their users because platforms, as well as various copyright holders on that platform, may differ in their opt-out preferences. As service owners, platform operators are also affected by the practices of third-party AI crawlers and may have to resort to blocking aggressive crawlers, notwithstanding any specific opt-out preference.

Considering the needs of software developers

The use of software code repositories such as GitHub is a widely established industry practice. Consequently, software developers are among the rights holders that routinely publish their works on third-party platforms they do not directly control. For software developers who wish to express their preferences regarding the collection of their works for AI training purposes, robots.txt is not an obvious or practical solution. Nevertheless, software code is a valuable input to AI training, not least demonstrated by the fact that AI-based coding assistants such as GitHub Copilot have been widely adopted in the industry. GitHub as the leading software development platform has an interest in facilitating a standard for expressing opt-outs that meet the needs of software developers.

To make opt-outs work for software code, it is important to consider the characteristics that distinguish it from other types of copyright-protected works. These characteristics include:

- **highly dynamic**: A work of software is rarely considered finalized, but rather undergoes a regular process of versioning, updating and improvement. To a lesser extent, this characteristic is shared by online news content, which may be dynamically updated after publication. Software code is even more dynamic due to the possibility to change comments or variable names, thus making significant changes to the text, without changing its function. Some emerging proposals for unit-based opt-outs, which try to automatically identify individual works or files using content-based identifiers or metadata, in order to subsequently remove them from training datasets irrespective of the location from which they were retrieved, are ill-suited for dynamic web content like software code;

- **directory-based**: Software code is often organized in a directory structure, including in versioned repositories (e.g., git) or archives (e.g., various package formats). While code hosting platforms offer the architecture for coding collaboration, software developers typically control the content of repositories. A standard for expressing opt-outs should allow software developers to express an opt-out within their repository, regardless of the specific architecture of the code hosting platform they use, to facilitate industry-wide adoption;
- **high prevalence of multiple authors**: Due to its collaborative nature, a work of software often has multiple authors, whose individual contributions can be difficult to disentangle. This raises the question of who should be entitled to declare an opt-out;
- **no collective management**: Unlike many other groups rights holders, software developers are not typically organized in collecting societies or other forms of collective representation;
- **frequent permissive licensing**: Open source software makes up a significant share of software code, with one study estimating the share of a given codebase originating from open source software to be as high as 77%. Open licenses typically preclude rights holders from introducing personalized contractual restrictions on re-use of openly licensed content. An opt-out within the meaning of Art. 4 EU CDSMD would therefore be incompatible with the use of a permissive license. Some open source components are re-used in thousands of other software repositories.

Considering these characteristics, we see significant barriers to the use of REP for the purpose of expressing opt-outs in the context of software code, most notably its reliance on a user's control over a domain and its size limitations (or control over page rendering in the case of Robots Meta Tags), and the risk of adverse effects on non-AI crawling purposes. Nevertheless, location-based opt-out mechanisms are more appropriate for dynamic content such as software code than unit-based approaches. GitHub would welcome the opportunity to contribute to the workshop in person to share the perspectives of platform operators and software developers.

Mike Linksvayer, VP, Developer Policy <mlinksva@github.com>
Felix Reda, Director, Developer Policy <felixreda@github.com>