AI Preferences
Internet-Draft
Updates: 9309 (if approved)
Intended status: Standards Track
Expires: 22 January 2026

G. Illyes
Google
M. Thomson
Mozilla

21 July 2025

Associating AI Usage Preferences With Content
draft-ietf-aipref-attach-02

Abstract

Content creators and other stakeholders might wish to signal their preferences about how their content might be consumed by automated systems. This document defines how preferences can be signaled as part of the acquisition of content in HTTP.

This document updates RFC 9309 to allow for the inclusion of usage preferences.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://ietf-wg-aipref.github.io/drafts/draft-ietf-aipref-attach.html. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-ietf-aipref-attach/.

Discussion of this document takes place on the AI Preferences Working Group mailing list (mailto:ai-control@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/ai-control/. Subscribe at https://www.ietf.org/mailman/listinfo/ai-control/.

Source for this draft and an issue tracker can be found at https://github.com/ietf-wg-aipref/drafts.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The automated consumption of content by crawlers and other machines has increased significantly in recent years. This is partly due to the training of machine-learning models.

Content creators and other stakeholders, such as distributors, might wish to express a preference regarding the types of usage they consider acceptable. Entities that might use that content need those preferences to be expressed in a way that is easily consumed by an automated system.

This document describes two mechanisms for associating preferences with content:

* A Content-Usage header field for HTTP [HTTP]; see Section 2.

* A Content-Usage directive for the Robots Exclusion Protocol (colloquially known as "robots.txt") [ROBOTS]; see Section 3.

For automated systems that use HTTP to gather content, these allow for the automated gathering of preferences in the same way that content is obtained.

1.1. Preference Expressions

The format of preference expressions is defined in the preference vocabulary [VOCAB]. The preference vocabulary defines:

* what preferences can be expressed,

* how multiple expressions of preference are combined, and

* how those preferences are turned into strings or byte sequences for use in a protocol.

This document only defines how the strings or byte sequences are conveyed so that the preferences can be associated with content.

1.2. Examples

A server that provides content using HTTP could signal preferences about how that content is used with the Content-Usage header field as follows:

```
200 OK
Date: Wed, 23 Apr 2025 04:48:02 GMT
Content-Type: text/plain
Content-Usage: train-ai=n

This is some content.
```

Alternatively, or additionally, a server might include the same directive in its "robots.txt" file:

```
User-Agent: *
Allow: /
Content-Usage: train-ai=n
```

1.3. Embedded Preferences

Embedding preferences is expected to be an effective means of associating preferences with content, because it ensures that metadata is always associated with content. This document, however, does not define any specific means of embedding preferences in content.

The main challenge with embedding preferences is that a different method might be needed for each content type. That is, a different storage or serialization model of conveying the preferences might need to be defined for each format whether it represent audio, documents, images, video, or other types of content. Furthermore, some content types, such as plain text (text/plain), offer no standardized means of embedding preferences.

The mechanisms in this document can be applied to any content type, provided that the content is obtained using HTTP (and maybe FTP). Future work might define how preferences might be indicated for alternative content distribution or acquisition methods, such as email.

1.4. Registry-Based Preferences

This document does not define a means of using unique identifiers and a registry for associating preferences. Registry-based approaches might be applicable in certain contexts, particularly where embedding is impractical or unavailable. Additionally, a registry might enable persistent association of preferences across distribution channels.

1.5. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. HTTP Content-Usage Header Field

The Content-Usage field is a structured field dictionary, as defined in Section 3.2 of [FIELDS]. This field follows the vocabulary and processing rules in [VOCAB].

This field indicates usage preferences regarding the content of the HTTP message. That is, the representation data, as defined in Section 8.1 of [HTTP], not the resource.

Servers MUST retain any preferences associated with a request if the content of that request is used to answer later requests. For example, the content of a PUT request that is used to answer subsequent GET requests. Note that servers that have not been updated to understand this field will not comply with this requirement.

The Content-Usage field does not have any special effect on caching.

3. Robots Exclusion Protocol Content-Usage Rule

The core function of Robots Exclusion Protocol format [ROBOTS] (or the "robots.txt" file) is to describe the expectations of the server operator about which paths can be crawled. This document adds a new rule that associates usage preferences with different paths. This new rule applies to any paths that can be crawled; paths that cannot be crawled have no associated usage preferences.

A Content-Usage rule is added to the set of potential rules that can be included in a Group for "robots.txt".

The rule ABNF pattern from Section 2.2 of [ROBOTS] is extended as shown in Figure 1.

```
rule =/ content-usage

content-usage = *WS "content-usage" *WS ":" *WS
                [ path-pattern 1*WS ] usage-pref EOL
usage-pref    = <usage preference vocabulary from [VOCAB]>
```

Figure 1: Extended robots.txt ABNF

Each group contains zero or more Content-Usage rules. Each Content-Usage rule consists of a path and a usage preference. The path might be absent or empty; if a path present, a SP or HTAB separates it from the usage preference.

Note that the usage preference expression encoding does not use an ABNF definition, relying instead on the definitions in [FIELDS].

3.1. Content-Usage Rule Semantics

Each group in the file applies to a set of crawlers, identified by product token as defined in Section 2.2.1 of [ROBOTS]. The Allow and Disallow rules determine what resources can be crawled, using the rule that has the longest matching path prefix, as defined in Section 2.2.2 of [ROBOTS].

This creates a two-stage arrangement that distinguishes acquisition and usage. Acquisition relies on Allow/Disallow rules; usage preference relies on Content-Usage rules.

Any Content-Usage rules determine the usage preferences for resources using the same path prefix matching rules as defined for Allow and Disallow. That is, the path prefix length is determined by counting the number of bytes in the encoded path.

Usage preferences apply only to those resources that can be crawled according to Allow/Disallow rules; no preferences are implied for resources that are disallowed.

Paths specified for Content-Usage rules use the same percent-encoding rules as used for Allow/Disallow rules, as defined in Section 2.1 of [URI]. In particular, SP (U+20) and HTAB (U+09) characters need to be replaced with "%20" and "%09" respectively.

The ordering of rules in a group carries no semantics. Thus, Content-Usage rules can be interleaved with Allow and Disallow rules.

If there are Content-Usage rules that have identical paths and conflicting usage preferences, these preferences apply separately according to the process defined in Section 7.1 of [VOCAB]. Note that this differs from the Allow/Disallow rules, where a conflict leads to the more permissive option, allowing crawling.

A crawlers can cache a "robots.txt" file for up to 24 hours, following HTTP Cache-Control semantics defined in [HTTP-CACHE]; see Section 2.4 of [ROBOTS] for details. Updates to preferences within the period that a file is cached might not be visible.

3.2. Processing Content-Usage Rules

To process a Content-Usage rule, a parser identifies lines with the "Content-Usage" label. This requires that SP and HTAB characters are ignored, before and after the label, in addition to before and after the COLON (":", U+3A) separator.

The remainder of the line - up to either the first CR (U+0D), LF (U+0A), or octothorpe ("#", U+23) - is the rule value.

The first character of the rule value will be "/" (U+2F) if a non-empty path is specified. Paths always start with a "/" character, so a rule value that starts with any other character indicates that the path is absent.

If a path is specified, the path ends immediately before the first SP (U+20) or HTAB ("U+09") character. The remainder of the rule value is the usage preference expression. If a path is absent, the entire rule value is the usage preference expression.

The usage preference is encoded using the exemplary format defined in Section 6 of [VOCAB]. The parsing and processing rules from Sections 6 and 7 of [VOCAB] apply.

Note that a usage preference expression is processed as a sequence of bytes, rather than Unicode text; see Section 6.3 of [VOCAB].

3.3. When Preferences Apply

A crawler that fetches resources uses the copy of "robots.txt" that is current at the time of the fetch to determine which usage preferences apply to those resources. Section 2.4 of [ROBOTS] defines how a "robots.txt" file can be cached.

This means that updates to "robots.txt" do not retroactively apply to resources. Changes to "robots.txt" that affect usage preferences therefore only apply after a crawler has retrieved the updated "robots.txt" and subsequently retrieved the affected resource again.

3.4. Example

Figure 2 shows a simple "robots.txt" document.

```
User-Agent: *
Allow: /
Disallow: /never/
Content-Usage: train-ai=n
Content-Usage: /ai-ok/ train-ai=y

User-Agent: ExampleBot
Allow: /
Content-Usage: train-ai=y
```

Figure 2: Example robots.txt file

A crawler that identifies as "ExampleBot" uses the second group. That crawler would be able to obtain all content and apply usage preferences of "ai=y" as defined in [VOCAB].

All other crawlers use the first group. This allows crawling of all content other than resources under "/never/". Of those resources, those under "/ai-ok/" have an associated usage preference of "train-ai=y" and all other resources have a usage preference of "train-ai=n".

Path	Crawl	Usage Preference
/test	yes	train-ai=n
/never/test	no	N/A
/ai-ok/test	yes	train-ai=y

Table 1: Sample of usage preferences for different paths

4. Security Considerations

Processing usage preferences involves the parsing of text that is produced by potential adversaries. Different guidelines for robust parsing can be found in Section 6 of [FIELDS] and Section 17 of [HTTP].

Section 3 of [ROBOTS] describes security considerations for "robots.txt". A "robots.txt" file can be up to 500KiB of text. This specification does not increase this limit.

5. IANA Considerations

The Content-Usage HTTP header field defined in Section 2 is added to the "HTTP Field Name" registry established in Section 18.4 of [HTTP]:

Field Name: Content-Usage
Status: permanent
Structured Type: Dictionary
Reference: Section 2
Comments: None

6. References

6.1. Normative References

[FIELDS] Nottingham, M. and P. Kamp, "Structured Field Values for HTTP", RFC 9651, DOI 10.17487/RFC9651, September 2024, <https://www.rfc-editor.org/rfc/rfc9651>.

[HTTP] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Semantics", STD 97, RFC 9110, DOI 10.17487/RFC9110, June 2022, <https://www.rfc-editor.org/rfc/rfc9110>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[ROBOTS] Koster, M., Illyes, G., Zeller, H., and L. Sassman, "Robots Exclusion Protocol", RFC 9309, DOI 10.17487/RFC9309, September 2022, <https://www.rfc-editor.org/rfc/rfc9309>.

[URI] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, DOI 10.17487/RFC3986, January 2005, <https://www.rfc-editor.org/rfc/rfc3986>.

[VOCAB] Keller, P. and M. Thomson, Ed., "A Vocabulary For Expressing AI Usage Preferences", Work in Progress, Internet-Draft, draft-ietf-aipref-vocab-02, 21 July 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-aipref-vocab-02>.

6.2. Informative References

[HTTP-CACHE] Fielding, R., Ed., Nottingham, M., Ed., and J. Reschke, Ed., "HTTP Caching", STD 98, RFC 9111, DOI 10.17487/RFC9111, June 2022, <https://www.rfc-editor.org/rfc/rfc9111>.

Acknowledgments

TODO acknowledge.

Authors' Addresses

Gary Illyes
Google
Email: garyillyes@google.com

Martin Thomson
Mozilla
Email: mt@lowentropy.net