

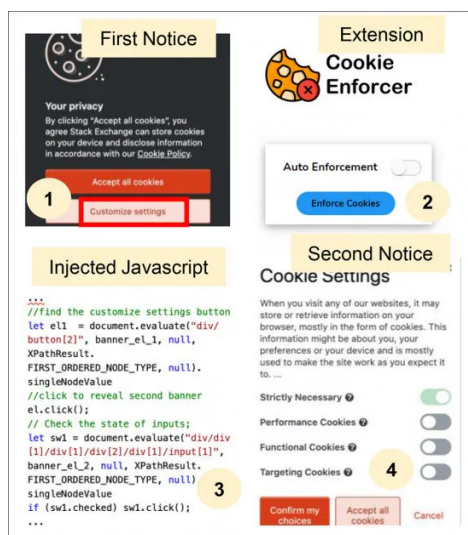
Machine Learning vs. Cookie Consent Systems

Originally published at <https://www.unite.ai/machine-learning-vs-cookie-consent-systems/>



A new research collaboration between the University of Wisconsin and Google sets machine learning against one of the most notorious web user annoyances of the last decade – the opacity and cynical misuse of GDPR-compliant cookie consent banners.

Titled *CookieEnforcer*, the new framework uses [Semantic Text Understanding](#) to parse the significance and utility of the underlying code behind the cookie consent popup or banner, in order to provide the user with the missing 'one click' solution to disabling all truly 'non-necessary' cookies – including the ones that domain owners may present as being 'essential', even if they are not.

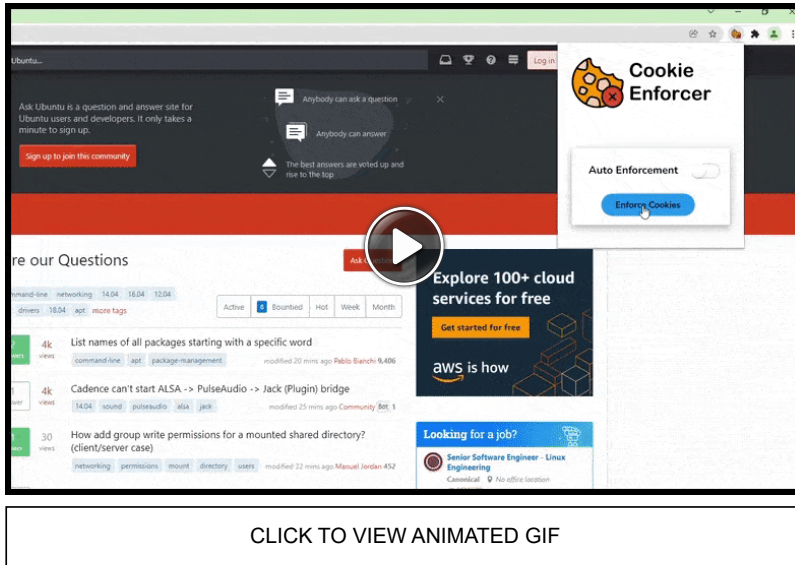


CookieEnforcer examines cookie consent code from the website www.askubuntu.com.

Source: <https://arxiv.org/pdf/2204.04221.pdf>

The system is implemented via a user-installed web browser plugin, which is capable of applying user-defined rules in a single click. Once a cookie consent framework appears on the website, the user can activate the plugin, which will then trawl the cookie consent code for potential actions before generating apposite JavaScript to enact choices on the user's behalf.

The plugin can be set to automatically enforce user preferences, or else take the cases individually, allowing the user to adjust settings before final submission.



Cookie enforcer in action. If preferred, the Chrome plugin can completely automate this process, without further user contribution. See later embedded video for more detail. Source: <https://www.youtube.com/watch?v=5NI6Q981quc>

The challenge of parsing the possible ‘non-consent’ options, which are typically hidden in arcane and laborious groups of settings (rather than the user-friendly *accept all* typical of consent frameworks) is modeled as a [sequence-to-sequence](#) task.

In an end-to-end accuracy evaluation, CookieEnforcer was able to generate all the necessary steps to obviate cryptic cookie consent procedures in 91% of the cases studied, on domains that had not been seen during training of the system’s machine learning model. A user study further demonstrated that the system significantly reduces user effort in navigating the consent modules.

The [paper](#) presenting the method is titled *CookieEnforcer: Automated Cookie Notice Analysis and Enforcement*, and comes from three researchers at the University of Wisconsin at Madison, and one from Google Inc.

Arcane Roads to Cookie Consent

Since the enactment of the General Data Protection Regulation ([GDPR](#)) in 2016 and the California Consumer Privacy Act (CCPA) in 2018, websites wanting to engage users from the areas covered by such legislation have been required to provide cookie preference mechanisms (usually based on detection of the user’s IP address as a proxy for their country of origin).

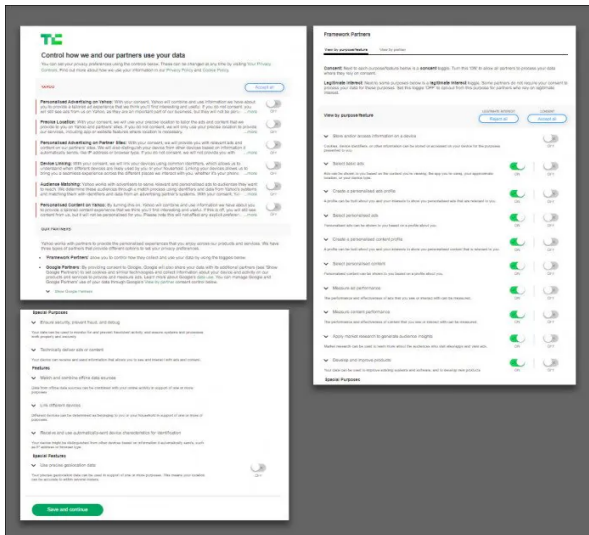
However, since domain owners had long been accustomed to gleaning valuable and actionable user data from the opaque and usually unseen implementation of cookies, they proved reluctant to furnish easy opt-outs for their newly empowered users.

The default UI for cookie consent interfaces (which appear the first time a user visits a domain, or if the user has deleted cookies for that domain) quickly settled into [dark patterns](#) designed to weary the viewer with granular, time-consuming, and extensive choices in the event that they wanted to exercise their rights to consent; or else a simple and easily accessible button which opted the user into all the cookies that the domain owner desired to run. This culture of labyrinthine UI choices was described in one 2020 study as [‘a scavenger hunt’](#).

The new paper comments:

‘[Users] may find it hard to exercise informed cookie control for websites with complicated notices. They are far more likely to rely on default configurations than they are to fine-tune their cookie settings for each [website]. In several cases, these default settings are privacy-invasive and favor the service providers, which results in privacy [risks].’

A comment on one [popular forum post](#) regarding these practices characterized them as ‘malicious compliance’. User annoyance with cookie consent frameworks is a topic that conflicts major publishers, who might ordinarily afford further coverage if they were not so personally exposed by their own practices in this regard.



A typical maze of options presented, in this case, by the TechCrunch website, ironically as a preface to an [article](#) on EU's changing attitude to what constitutes cookie consent. The appended URL identifiers and hooks designed to further enable tracking stood at 262 characters (deleted here). A 'reject all' button, while available for certain categories of cookie, is not available for the entire set of possible cookies; in those excepted cases, the user must operate each 'toggle'.

A [2019 paper](#) from Germany found that a majority of site visitors in the studied domains were 'nudged' towards broad consent, and that only a third of websites actually explained the purposes of the data collection practices.

A number of web browser plugins, add-ons and extensions have emerged to address the problem in recent years, such as the [Cookie Quick Manager](#) Firefox extension, and a broad range of Chrome alternatives, while the European Union is [seeking to close up](#) the compliance loopholes around cookie consent architectures.

Method and Data

The researchers of the new paper were determined to create a more robust cookie consent management framework by avoiding reliance on keywords or handcrafted rules, the central approach of a number of recent similar [ML-aided projects](#).

CookieEnforcer has three objectives: to translate cookie notices and interfaces into a machine readable format; to identify the cookie setting configuration in a manner that disables non-essential cookies; and to automatically apply additional restrictions without further user input, if desired by the user.

The system consists of a backend component that detects and analyzes cookie notices, and a frontend component, in the form of a browser extension, that generates and executes the disabling of non-essential cookies (i.e. cookies that will not obstruct navigation of or access to the domain if blocked).

The framework is embodied in a Chrome-specific locally installed extension which uses the [Selenium](#) web testing library under the [ChromeDriver](#) framework.

The backend section features modules for detection, analysis, and a decision model. The analysis module takes account of changes in code introduced by user interaction, so that the initial code dump is not rendered invalid by simulated user exploration.

Natural Language Understanding

With the code revealed, it's important that CookieEnforcer understand the existing state of possible actions it might take, since the language behind toggle buttons can be ambiguous in terms of benefit to the end user.

To this end, the researchers trained a [Text-To-Text Transfer Transformer](#) (T5) model for its decision component. The T5-Large model, which contains 770 million parameters, was fine-tuned on a custom database of input/output code (i.e., code that describes and enables the functionality of toggling options).

Input-Output format for the Decision model.	
Input:	< notice.0.tag.0> - <notice.0.tag.0.text>, <notice.0.tag.0.state> <notice.0.tag.1> - <notice.0.tag.1.text>, <notice.0.tag.1.state> - ... ** <notice.1.tag.0> - <notice.1.tag.0.text> - ... <end>
Output:	Click <notice.0.tag.0> Click <notice.0.tag.2> ** Click <notice.1.tag.2>

Example of a training sample.	
Input:	button0 - customize settings button1 - accept all cookies ** switch3 - performance cookies, not selected switch4 - functional cookies, not selected switch5 - targeting cookies, not selected button6 - confirm my choices button7 - accept all cookies button8 - cancel <end>
Output:	Click button0 ** Click button5.

Sample formatting (above) and training data (below) for the T5 model. The data example is from www.askubuntu.com.

The dataset was created by sampling 300 websites with cookie notices selected from Tranco's [top-50k popular websites](#) list. The detector and analyzer modules extracted the cookie consent options from their runtime source code, and evaluated their default states.

One of the researchers then manually labeled the interpreted series of clicks necessary to disable non-essential cookies for all the studied websites, resulting in 300 fully labeled domains.

Website	Input	Output
www. reddit. com	button0 - reject non-essential button1 - accept all <end>	Click button0.
www. netflix. com	button1 - learn more about our use of cookies and information. button4 - accept button5 - reject button6 - personalise my choices button7 - close ** button0 - close switch5 - advertis- ing cookies , not selected button27 - save settings <end>	Click button5.
www. wordpress. com	button0 - customize button1 - accept all ** switch3 - analytics: these cookies allow us to optimize performance by collecting , selected switch4 - advertising: these cookies are set by us and our advertising , not selected button5 - accept selection <end>	Click button0 ** Click switch3 Click button5.
www.tata. com	button0 - sweet! button1 - sorry, i'm on a diet <end>	Click button1.
www. newscientist com	button1 - i accept button2 - show purposes ** button4 - select basic ads; object to legitimate interests ... switch23 - analytics cookies , not selected button62 - confirm my choices	Click button2 ** ... Click button4 Click button62.

Variety in source code disposition across examples from the custom dataset.

60 websites were set aside as a test set, and the T5-Large model was trained with a learning rate of 0.003 at a batch size of 16 for 20 epochs, with a maximum input sequence length of 256 tokens, and a maximum target sequence length of 64. The tokens were formed of sub-words established by Google's [SentencePiece](#) tokenizer.

Finally, the processed information is stored in a local database and made available to the front end of the system. The authors favored the [querySelector\(\)](#) HTML function over the XML Path Language ([XPath](#)) approach taken by some previous similar projects, since XPaths for cookie notices are vulnerable to DOM updates (i.e. the code may change after initial loading in response to user interactions). In this way, the element paths can be retained even when they are dynamic and responsive to external factors.

Testing and Performance

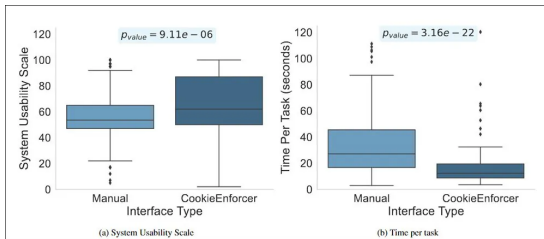
In practice, CookieEnforcer proved able to navigate some of the darkest dark patterns in the dataset, such as a hidden option in the cookie consent framework of *The New Scientist* which is obscured by JavaScript until the user explicitly requests to see it.

The authors comment:

'This option can be easily missed by the users as they have to expand an additional frame to see that. CookieEnforcer not only finds this option, but also understands the semantics and decides to object. These examples showcase that the model learns the context and generalizes to new examples.'

The researchers performed three tests, including an end-to-end evaluation of the framework's performance across 500 unseen domains (i.e. websites that CookieEnforcer was not specifically trained for), where the authors report that it could successfully disable non-essential cookies for 91% of the sites.

The second test comprised an online user study spanning 14 websites, and using the System Usability Scale (score) against a manual baseline. For this test, the authors report that CookieEnforcer obtained a 15% higher score than the baseline.



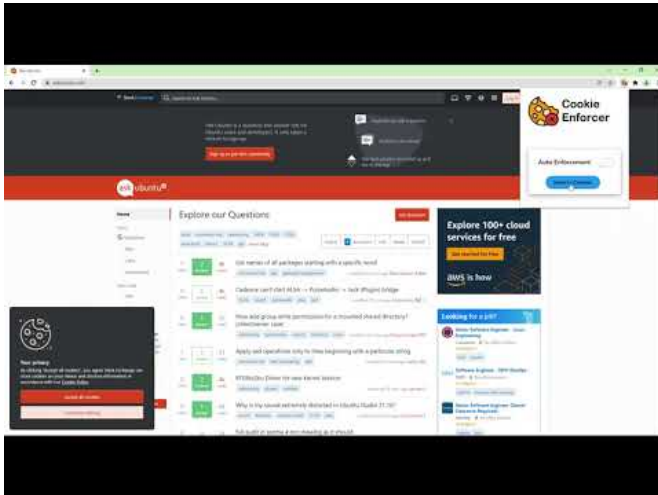
CookieEnforcer enables a 15% higher score than baseline (non-aided) usage, at the same time automating a vexing process.

Finally, CookieEnforcer's trained parameters were tested against the top 5000 websites in the US and Europe, to determine its capacity to navigate cookie notices. The authors state:

'While measurements at such a scale have been performed before, CookieEnforcer allows a deeper understanding of the options beyond keyword-based heuristics. In particular, we find that 16.7% of the websites in the UK showing cookie notices have enabled at least one non-essential cookie. The same number for websites in the US is 22%.'

The authors have released a short YouTube video showing CookieEnforcer in action:

Cookie Enforcer disabling non-essential cookies on askubuntu.com



First published 12th April 2022.

Writer on machine learning, domain specialist in human image synthesis. Former head of research content at Metaphysic.ai, until its dissolution into DNEG's Brahma.ai.

Portfolio site: martinanderson.ai

Contact: martin@martinanderson.ai



Discover More