

Designing standardized pluggable recommender systems

Increasingly, government proposals for tech regulation include calls to require online platforms to provide users with a choice of presentation algorithms for contexts such as search results and social media feeds. For example, the Digital Services Act as adopted by the European Parliament requires “very large online platforms” to offer a choice of recommender systems where at least one option “is not based on profiling.”¹ The Filter Bubble Transparency Act introduced in both the U.S. House and Senate in 2021 with bipartisan support requires platforms to provide a choice of what it calls an “input-transparent algorithm” if its default presentation is “opaque” and derived from “user-specific data.”² Currently, these bills presume that all options for recommender systems would be operated by the platform service provider, and of course that party has the relevant information necessary for optimization in some dimensions. However, continued momentum for interoperability mandates in legislative proposals³ could upend these assumptions and lead to a technology landscape where online platforms are required to allow their users to choose third-party recommender systems to filter and prioritize the content to which they have access through the platform.

In parallel to these developments, many active projects are pursuing the development of standardized protocols in specific functional areas such as messaging and social networking, building on existing widely-used protocols such as XMPP and ActivityPub. However, the notion of user-configurable, third-party recommender systems remains novel, and merits consideration. To maximize the benefits of user choice and interoperability, the functionality of a recommender system should be standardized and able to be developed by a third party to work with an existing online platform. Functioning as an intermediary operating between the user/client software and the background service/infrastructure, such pluggable recommender systems – able to be chosen by a user at will with robust multihoming capabilities, like a filter on a photography app – would greatly increase the perception and reality of individual agency with respect to online services and would help promote a diversity of online experiences.

Standard APIs or protocols are therefore desirable to shape the emergence of such an ecosystem from the outset. Optimally, both the input to the recommender system (i.e., data to

¹ See Amendment 330 to DSA Article 29, paragraph 1, available at:

https://www.europarl.europa.eu/doceo/document/TA-9-2022-0014_EN.html (“1. In addition to the requirements set out in Article 24a, very large online platforms that use recommender systems shall provide at least one recommender system which is not based on profiling, within the meaning of Article 4 (4) of Regulation (EU) 2016/679, as well as an easily accessible functionality on their online interface allowing the recipient of the service to select and to modify at any time their preferred option for each of the recommender systems that determines the relative order of information presented to them.”).

² Sec. 3 of S.2024 (2021) and the identical H.R. 5291 (2021) require both notice and provision of “a version of the platform that uses an input-transparent algorithm and enables users to easily switch between the version of the platform that uses an opaque algorithm and the version of the platform that uses the input-transparent algorithm by selecting a prominently placed icon, which shall be displayed wherever the user interacts with an opaque algorithm.”

³ For example, S.2710 as adopted Feb 3, 2022 by the Senate Judiciary Committee (by a lopsided 21-1 vote) includes a section dedicated to interoperability, requiring operating system vendors to permit users to install and use third-party apps and app stores.

and from the online platform) and its output (to and from client software) would be standardized through public, well documented APIs; at minimum, the input layer must be provided.

Rather than solely focused on interoperability among horizontally positioned services that offer comparable functionality within known domains such as messaging, a pluggable recommender system infrastructure would create incredible new opportunities for innovation, competition, and independent user choice leading to a rich diversity of online experiences.

Such a framework raises many technical challenges worth investigating, including:

1. Efficiency/latency - for a data-rich service, the recsys must retrieve a high volume of data from the platform service provider. This could be achieved in many circumstances through background retrieval and caching, APIs permitting; alternatively, providing more efficiency but perhaps less control, the data could remain solely with the platform and predetermined configuration levers would be exposed to the recsys.
2. Privacy - sharing platform data with a third-party service has raised privacy concerns since before Cambridge Analytica; the greater force of data protection regulations are not a sufficient answer, and some combination of law, contract, and auditing is needed.
3. Monetization - to the extent this intermediation deprives the platform of some forms of revenue, particularly advertising, some new approaches may be needed. Contractually forcing the display of advertising may work in some circumstances, but in others, wholesale pricing models may need to develop to pass costs from the platform provider to the recsys operator.

