

Improving Web Interactions with Machine Learning

Corin R. Anderson • Pedro Domingos • Daniel S. Weld

The Web for the masses



Same page for everyone

Most web sites are designed following a one-size-fits-all philosophy – that the same content and site structure is appropriate for all visitors. But, this assumption is often not true. Users with different needs to fulfill look for different content (e.g., looking for RAM prices versus looking for ultralight notebook reviews). And, visitors' navigational styles vary widely (some prefer to drill down a hierarchy of links, others use search forms, etc.).



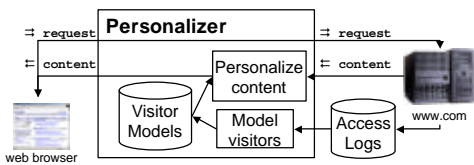
Same page every visit

Even more sites on the web assume that visitors want to see the same content on each visit. However, this is often not the case. After a visitor has read a series of news stories, she does not want to read them again. When a user views her portal page in the morning, she may be interested in stock prices, but in the afternoon, she's more concerned about traffic information.

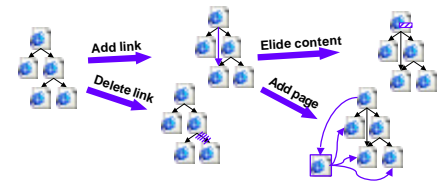
The Web for you

Web sites should adapt and customize themselves automatically for visitors. Content should be **dynamic** to the browsing context and **personalized** for each visitor.

The Proteus architecture



Web usage log	Time	Path
some-pc.cs	22/Feb/2001 11:49:13	/index.html
some-pc.cs	22/Feb/2001 11:49:23	/index.html /hardware/
some-other-pc.cs	22/Feb/2001 11:49:55	/index.html
some-other-pc.cs	22/Feb/2001 11:50:20	/index.html /reviews/
some-pc.cs	22/Feb/2001 12:12:36	/index.html /hardware/notebooks



Web site personalizers

Situated between the web site and the visitor, a personalizer adapts and customizes the content as the visitor browses. The personalizer builds a model of the visitor based on access logs, either from the web site or from the personalizer itself. This model is used to evaluate alternative personalizations.

Step 1: Model visitors

We view personalization as a two-step process. In the first step, the personalizer builds a model of the visitor based on web usage logs. The model contains information about navigational behavior – which pages the visitor views – as well as the content of the pages – what keywords interest the visitor.

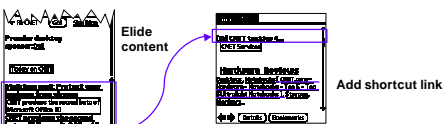
Step 2: Maximize utility

In the second step, the personalizer customizes the site to maximize the benefit the visitor receives by viewing the content – the expected utility. Beginning with the unmodified site, the personalizer searches through the space of alternatively-personalized web sites to find the one with the greatest expected utility.

Implemented systems

Proteus

Following the architecture above, we implemented a personalizer for wireless web visitors: **Proteus**. Proteus adds shortcuts to, and elides content from, each requested page. Proteus measures expected utility by summing the value of each page in the site discounted by how hard it is to reach each page. In a study to evaluate Proteus, users required less time and fewer navigational actions to find information on the personalized sites than on the original ones. **Proteus demonstrates the value of personalization.**

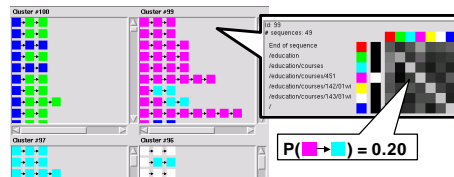


Proteus in action

The screenshots on the left show cnet's main entry page without personalization. The screenshot above shows how Proteus has personalized the page for a mobile visitor.

MinPath

Our initial implementation of Proteus modeled each visitor in isolation. However, particularly when data about the visitor is sparse, these models may be inaccurate. To do better, we developed **MinPath**, an algorithm that combines data from many, similarly-behaving visitors to guide how shortcuts are selected.



Clustering web visitors

MinPath identifies groups of visitors by clustering their sequences of page requests, and builds models of each of these clusters. MinPath uses these models to predict which links the visitor will follow and hence how useful a shortcut may be. In experiments on data from the UW CSE web site, **MinPath saved 40% of the unnecessary navigation** for visitors.

Montage

The Proteus architecture can also be used to build personalized, dynamic portal sites. On an internship with Eric Horvitz at Microsoft Research, we developed the Montage system that builds dynamic start pages. Montage selects content based on users' past interest in pages and on the user's current browsing context (time of day, topic of recent browsing, etc.). Montage builds a portal site, including a Main Montage and montages for specific topics. Each montage includes links to relevant pages and embedded "lenses" of distal pages that are of most interest to the user.



For more information

www.cs.washington.edu/homes/corin/

corin@cs.washington.edu