Auto–Personalization Web Pages

Jon T. S. Quah
Nanyang Technological University, Singapore

Winnie C. H. Leow
Singapore Polytechnic, Singapore

K. L. Yong
Nanyang Technological University, Singapore

INTRODUCTION

This project experiments with the designing of a Web site that has the self-adaptive feature of generating and adapting the site contents dynamically to match visitors' tastes based on their activities on the site. No explicit inputs are required from visitors. Instead a visitor's clickstream on the site will be implicitly monitored, logged, and analyzed. Based on the information gathered, the Web site would then generate Web contents that contain items that have certain relatedness to items that were previously browsed by the visitor. The relatedness rules will have multidimensional aspects in order to produce cross-mapping between items.

The Internet has become a place where a vast amount of information can be deposited and also retrieved by hundreds of millions of people scattered around the globe. With such an ability to reach out to this large pool of people, we have seen the expulsion of companies plunging into conducting business over the Internet (e-commerce). This has made the competition for consumers' dollars fiercely stiff. It is now insufficient to just place information of products onto the Internet and expect customers to browse through the Web pages. Instead, e-commerce Web site designing is undergoing a significant revolution. It has become an important strategy to design Web sites that are able to generate contents that are matched to the customer's taste or preference. In fact a survey done in 1998 (GVU, 1998) shows that around 23% of online shoppers actually reported a dissatisfying experience with Web sites that are confusing or disorganized. Personalization features on the Web would likely reverse this dissatisfaction and increase the likelihood of attracting and retaining visitors.

Having personalization or an adaptive site can bring the following benefits:

1. Attract and maintain visitors with adaptive contents that are tailored to their taste.

2. Target Web contents correspondingly to their respective audience, thus reducing information that is of no interest to the audience.
3. Advertise and promote products through marketing campaigns targeting the correct audience.
4. Enable the site to intelligently direct information to a selective or respective audience.

Currently, most Web personalization or adaptive features employ data mining or collaborative filtering techniques (Herlocker, Konstan, Borchers, & Riedl, 1999; Mobasher, Cooley, & Srivastava, 1999; Mobasher, Jain, Han, & Srivastava, 1997; Spiliopoulou, Faulstich, & Winkler, 1999) which often use past historical (static) data (e.g., previous purchases or server logs). The deployment of data mining often involves significant resources (large storage space and computing power) and complicated rules or algorithms. A vast amount of data is required in order to be able to form recommendations that made sense and are meaningful in general (Claypool et al., 1999; Basu, Hirsh, & Cohen, 1998).

While the main idea of Web personalization is to increase the 'stickiness' of a portal, with the proven presumption that the number of times a shopper returns to a shop has a direct relationship to the likelihood of resulting in business transactions, the method of achieving the goal varies. The methods range from user clustering and time framed navigation sessions analysis (Kim et al., 2005; Wang & Shao, 2004), analyzing relationship between customers and products (Wang, Chuang, Hsu, & Keh, 2004), performing collaborative filtering and data mining on transaction data (Cho & Kim, 2002, 2004; Uchyigit & Clark, 2002; Jung, Jung, & Lee, 2003), deploying statistical methods for finding relationships (Kim & Yum, 2005), and performing recommendations bases on similarity with known user groups (Yu, Liu, & Li, 2005), to tracking shopping behavior over time as well as over the taxonomy of products. Our implementation works on the premise that each user has his own preferences and needs,

and these interests drift over time (Cho, Cho, & Kim, 2005). Therefore, besides identifying users' needs, the system should also be sensitive to changes in tastes. Finally, a truly useful system should not only be recommending items in which a user had shown interest, but also related items that may be of relevance to the user (e.g., buying a pet => recommend some suitable pet foods for the pet, as well as suggesting some accessories that may be useful, such as fur brush, nail clipper, etc.). In this aspect, we borrow the concept of 'category management' use in the retailing industry to perform classification as well as linking the categories using shared characteristics. These linkages provide the bridge for cross-category recommendations.

DESCRIPTION OF SYSTEM

In this article, we seek to provide an adaptive feature using a fast and cost-effective means. The aim is to provide adaptiveness in the sense that when a visitor selects the next link or a new page, the contents of the page generated will have relatedness to previous pages' contents. This adaptive feature will be immediate and will not experience delay or repetitive computational filtering problems, as compared to using mining or collaborative filtering (Claypool et al., 1999; Basu et al., 1998).

The rules-based technique offers an excellent and flexible mechanism to specify rules that map categories that exhibit relatedness among themselves (IBM, 2000). Adding new product lines is simple, by just adding new sets of rules to map the new products accordingly. For direct item-to-item relatedness mapping, it is not so scalable and feasible to implement through use of the rules-based technique. Instead we will use content-based filtering for generating direct item-to-item mappings. The content-based technique (Claypool et al., 1999; Basu et al., 1998) allows item-to-item mapping to be implemented in a scalable manner by just defining the item's attribute, and the recommendation engine will automatically generate or match items of same attribute without involving user efforts (Basu et al., 1998).

In order to facilitate the deployment of these recommendation techniques, the Web domain is structured into their respective categories that exhibit relatedness among them. For example, pet dog would have relatedness to books on dogs. Each of the categories is given a unique ID value. The relatedness rules make use of these IDs to generate recommendations. The Web site domain is structured into supernodes (SA, SB…) which branch into child nodes (A1,A2…An;…K1,K2…Kn). These supernodes are a representation of products on the Web site, and the child nodes are used to represent the breakdown of the products into categories. Below each of the child nodes are the sub-child nodes (Aa1,…Axm,…Ka1,Ka2,…Kxm) that represent the items. Each of the child nodes (A1,A2…An;…K1,K2…Kn) is identified with its corresponding ID value. With this structure, rules-based mapping can be easily identified and applied among the child nodes by defining the IDs that will result into a recommended page.

The syntax of a relatedness rule is:

IDa:IDb:…=>Target page

The entries IDx represent the IDs of Web pages that have relatedness and thus can be mapped directly to the Web page (link) identified as target page. A rule is considered matched when any of the Web page IDs in the rule is also found to exist in the selector's list (visitor's profile). The selector's list is compared against all the rules in the rule file. Only one of the rules will be used, and that is the rule that has the most number of IDx elements matching the selector's list. In the event of tie, the rule which matches with selector entry that carries the higher points will be used, or if they still tie, then precedence of rule entry in the rule file will be used to determine the final rule to use.

Mappings across sub-child nodes are done based on their attributes (content based). A file will be used to define the list of items (or contents) of each Web page (at category level) and also any of attributes for each of the items. The syntax for this item list file is:

Idxa:idxb:…:attra:attrb=>display.html

The entries idxx are indexes to the html page that contains all the information regarding the item. An example of these indexes could be the type of pet. The information about each type of pet is found in the corresponding html file. An item sorter engine will, at startup, sort out the indexes and their corresponding html files. This will allow the page contents to be arranged or displayed based on any of these indexes. The other set of entries attrx define the attributes of the item. When the visitor selects any of the items for browsing, the corresponding attrx will be tracked. Whenever the visitor browses a new page, the recommendation engine will check for items that have the same attrx and automatically include the corresponding html file for display (see Figure 1).

Our prototype system also incorporates means of generating a dynamic profile that changes as the visitor browses through the Web site. Implicit tracking does not require any explicit inputs or intervention by the visitor. Conventionally this is done either through use of user authorization, hidden form fields, URL rewriting, or cookies (Hunter et al., 2002). Although cookies offer an elegant

Related Content

Reverse Logistics in E-Retaling
Senem Erganand Aye Akyol (2021). *Research Anthology on E-Commerce Adoption, Models, and Applications for Modern Business (pp. 1001-1027).*
www.irma-international.org/chapter/reverse-logistics-in-e-retaling/281546

The Internet Moderating Impact among Stakeholders in Modern Organizations: A Conceptual Analysis of Service Management Environment
Mohammad Ayub Khanand Khalid S. Soliman (2009). *Journal of Electronic Commerce in Organizations (pp. 21-30).*
www.irma-international.org/article/internet-moderating-impact-among-stakeholders/37397

Framework for User Perception of Effective E-Tail Web Sites
Sang M. Lee, Pairin Katerattanakuland Soongoo Hong (2005). *Journal of Electronic Commerce in Organizations (pp. 13-34).*
www.irma-international.org/article/framework-user-perception-effective-tail/3448

Service Encapsulation-Based Model for Smart Campus
Ying Chen, Runtong Zhangand Shouyi Zhang (2012). *Journal of Electronic Commerce in Organizations (pp. 31-41).*
www.irma-international.org/article/service-encapsulation-based-model-smart/72998

The TradeCard Financial Supply Chain Solution
Soe-Tsyr Yuan (2007). *International Journal of Cases on Electronic Commerce (pp. 48-70).*
www.irma-international.org/article/tradecard-financial-supply-chain-solution/1509