**CHAPTER I**

**INTRODUCTION**

User Experience, or UX, is a very important part of web design. User Experience is the process of enhancing customer satisfaction and loyalty, by improving the usability, ease of use, and pleasure provided in the interaction between the customer and the product. A major part of User Experience of a Web Browser, is when customers using it to browse the Internet, is when the browser appears tailor-made for their needs. To emulate such a browser for maximum user satisfaction, extensive research has been done to orient aspects of a user's browsing experience to his exact needs.

Recent popular research on online browsing behavior[1], data conducted through studies using a new Yahoo! Toolbar yielded a variety of features affecting user's viewing preferences, including extent of page visits, time spent on a page, burstiness of pageviews associated with a URL, and mechanism of visiting and traversing through pages. A taxonomy of classes can be created, to characterize pages into Content, Communication and Search. Studies show which factors do affect browsing behavior and page ranking, such as method of reaching the page, and also which factors are insignificant, such as burstiness for a site with multiple users.

The method of browsing can also be broken down to analyse its significance in the personalization process. The process generally contains three components, Web Services, Proxy Server, and the Client. Recommendation algorithms are among the most effective methods for personalization, and may be applied among these process stages to optimize browsing experience.

In this project, we have presented various methods for improving browser behavior. The first half of the project is concerned with analyzing user activity and providing easy access to browser top sites as well as future website recommendations to a user. The second half of the project proposes a simple chrome-extension based test that classifies users using Decision trees into three categories: Beginner, Intermediate and Expert. We further customize a user’s browsing experience based on their skillset. Hence, this project was concerned with optimizing browser experience for a user by focusing on a user’s preferences, skills and browsing activities. The next chapter (Chapter II) provides the details of a thorough literature survey that we conducted. Chapter III and IV discuss the two major aspects of this project. The project seemed to provide great results that will be discussed in Chapter V. The project is concluded in Chapter VI.

**CHAPTER II**

**LITERATURE SURVEY**

**1. A Characterization of Online Browsing Behavior[1]**

**Authors:** Ravi Kumar (Yahoo!) and Andrew Tomkins (Google)

**Publication Details:** WWW '10 Proceedings of the 19th international conference on World wide web; ACM 2010.

**Summary of Abstract:** A large-scale study of online user behavior is undertaken, based on search and toolbar logs, and a new CCS taxonomy of pageviews consisting of Content (news, portals, games, verticals, multimedia), Communication (email, social networking, forums, blogs, chat), and Search (Web search, item search, multimedia search) is proposed. It is shown that roughly half of all pageviews online are content, one-third are communications, and the remaining one-sixth are search. Further breakdowns are given to characterize the pageviews within each high-level category. The extent to which pages of certain types are revisited by the same user over time is studied, and the mechanisms by which users move from page to page, within and across hosts, and within and across page types. Robust schemes for assigning responsibility for a pageview to ancestors along the chain of referrals are considered. Finally, the burstiness of pageviews associated with a URL is studied.

**Summary of Methodology:** Data iss collected from from Yahoo! Toolbar users over a period of one week. The amount of time spent online is estimated, and analyzed. Inter-arrival time between websites is also studied. Websites are canonicalized to analyse number of pageviews for each category of websites. A Content, Communication and Search taxonomy of pageviews is then defined after analysis. Automated analysis of page types is carried out and it is tuned for high precision and attained recall. Session reuse, i.e. re-finding a previously found URL, is observed and it was seen that portals scored higher. Referrals, i.e. how the user navigated to the site, are found to significantly affect browsing behavior. Burstiness is recorded and it does not affect browsing behavior. Search behavior is defined as search trees and it is also analyzed.

**Summary of Results:** Taxonomy of classes is proposed, Content, Communication and Search. Mail, news and social networking appear in homogeneous sessions of one type. Search pageviews appear on the path to a disproportionate number of pageviews, but search cannot be considered as the principal mechanism of reaching the pageviews. Online browsing behaviour is not significantly affected by burstiness .Longer tree-represented search sessions correspond to more time spent engaging off the search engine with content on the rest of the Web.

**2. Recommendation-Assisted Personal Web[2]**

**Authors:** Haiming Wang and Kenny Wong

**Publication Details:** IEEE 9th World Conference on Services (IEEE 2013)

**Summary of Abstract:** With the growing establishment of Internet infrastructure, more and more online services become available to end user, which in turn promotes the prosperity of the Internet. However, two issues emerge during this information increase. First, users have to access many individual sites to get their services, which consume lots of time and contains some duplicate work. Second, user traces in different websites could have been used to provide more personalized services. Given these observations, this position paper proposes a recommendation assisted personal web system based on existing work on personal web and recommendation systems. This system can integrate several web services to form a personal web, derive request specific user data, and provide a personalized service by content based filtering and user intention inference. Using a research assistant application as a case study, we show how this framework helps to deliver personalized services.

**Summary of Methodology:** There are three main components involved in system design of the web browsing procedure, Web Services, Proxy Server and Client. Web Services module provides information or services, using personalization module. Proxy Server is intermediary between user and web services. The three process phases are Profiling, Request Modelling and Forwarding, Response Manipulation, and Interaction. A case study is conducted, for a recommendation-assisted personal assistant. Data model consists of the dataset and recommendation algorithms. Semantic and social network analysis is applied on the Data Model layers. Latent Dirichlet Association is applied to extract topic distribution of each document, and community mining algorithms to cluster nodes by their links. Recommendation algorithms are formulated accordingly, and weighted. The ﬁnal user selection is stored in the query history of the user model, making the system adaptive.

**Summary of Results:** It is observed that incorporating domain knowledge is important. If no domain knowledge is available, the approach changes to query expansion. Collaborative filtering can be adopted if number of users is high. Search engines help to index the web, but the search results could be more optimized towards user interests. Recommendation system approaches are used to personalize a web service. Integration of several web services and application in areas that already have web services and provide a domain-specific personalized service can be done.

**CHAPTER III**

**HISTORY – BASED DASHBOARD**

**3.1 Analysis of History**

A basic Chrome extension is built, and history API is used to extract browser history of a user. The history dataset collected is then cleaned by removing all query strings from the URLs, while only maintaining the base URL. The frequency of each URL is calculated, and a weighted sum of the visit frequency, time spent and recency of last visit, is used to re-order the history dataset in terms of user preference, and a range of URLs is chosen. The chosen most popular URLs from the history are categorized by scraping category data from the Internet, using a basic parsing code. The scraped dataset, too, is cleaned, and transferred into the control of the extension dashboard.

**3.2 Dashboard Creation**

The dashboard consists of a Search option, as well as a dynamic and categorized display of the user's favorite pages. To increase customization value and ease of use, the dashboard features are smoothly and seamlessly integrated into the dashboard interface. Page categories are typically social network, search, entertainment and so on. Each feature has its own display tab, with the options to view these features by selecting the appropriate tab. Screenshots of each of the user's visited pages are taken to be used as icons for the dashboard items, however, this approach requires accumulation of a large number of potentially unnecessary screenshots. Instead, logos for the suggested websites on the dashboard page are loaded when necessary. The dashboard also includes a’ recommended sites to visit’ section.

**3.3 Similarity** **Estimation**

In order to build a recommendation system for websites according to previously visited websites, a database of websites is required. A list of 100 most popularly used sites is obtained, and short textual descriptions for these sites are scraped from the Internet to build a base dataset for the similarity algorithm. Unigram models are constructed from the descriptions, and stop words are eliminated. Similarity between each pair of URLs is calculated, using a cosine similarity formula for the significant words in the description of the website as found. An upper triangular similarity measures matrix is constructed, using the cosine similarity formula. Since the generated matrix is too large to use in the program, a list of ten websites with highest similarity scores is maintained for each website. These top ten websites were recommended to a user whenever some website is prioritized enough by the formula used to rank website preference from the user's history.

**3.4 Recommendation Visualization**

A Word Cloud is used to visualize the results of the recommendation algorithm. Word Cloud is chosen to represent the results, because considering the user-friendly intention of the application, it is the best visualization tool for the purpose. Sites which are highly recommended are highlighted in the Word Cloud by displaying them in larger font sizes, and sites with lesser recommendation are made smaller, to visually indicate the same. Each website entry in the Word Cloud also doubles as a hyperlink to the page in mention by the entry. Thus, it is a suitable tool for all calibers of users, due to its aesthetic appeal and its ease of interpretation and use.

**CHAPTER IV**

**CLASSIFICATION OF USERS BASED ON FAMILIARITY WITH BROWSING FUNCTIONS**

**4.1 Goal**

In this part of the project, we have aimed at customizing a user’s browser based on his expertise with it. People who have only been introduced to browsers recently often struggle to understand its features. Hence, in this section of the project we have decided to classify users based on their expertise level by providing them with a chrome extension based test. This test is adequately designed to measure the skill level of users. Based on the classification of a user that is obtained after the test, the user’s browser homepage is automatically customized.

**4.2 Classification Classes**

1. Beginner: Users that are very new to the browser and can usually only perform minimal use

2. Intermediate: Regular users who have gotten accustomed to some common browser features but do not understand any of the other uncommon but essential features

3. Expert: Due to their extensive use of browser, they now understand almost all important browser features

**4.3 Platform – Chrome Extension**

Extensions allow you to add functionality to Chrome without diving deeply into native code. The most important file required for the extension to function, is the manifest.json file. This manifest is nothing more than a metadata file in JSON format that contains properties like the extension's name, description, version number. Chrome comes packed with a lot of developer tools, one of which is the Chrome Extension API set. The use of APIs to access browser data can raise certain security issues, hence appropriate permissions are required to access the data. These permissions are also part of the manifest.

**4.4 Browser Behaviour Test for Training Data Collection**

The training data is collected by creating a quiz which tests the familiarity of the user with the functionality of the browser. The data is collected from users belonging to the age group 15-60.

**4.4.1 Browsing Tasks**

1. Adding a Bookmark

API used – chrome.bookmarks

Permission - bookmarks

2. Creating a Bookmark Folder

API used – chrome.bookmarks

Permission - bookmarks

3. Changing the Font Size

API used – chrome.fontSettings

Permission - fontSettings

4. Changing the following Privacy Settings:

API used – chrome.privacy

Permission - privacy

a. Autofill Settings

b. Spelling-Error Detection

c. Third-Party Cookie Settings

d. Password Saving

e. Translation Services

5. Removing a Bookmark

API used – chrome.bookmarks

Permission – bookmarks

6. Deleting a History Item

API used – chrome.history

Permission - history

The time taken by the user to perform the above tasks is timed individually. The user action is tracked and on completion of a particular task, the next task is displayed. The chrome.tabs API is used to switch the browser tabs, and requires the permissions “tabs” and “activeTab”.

**4.4.2 User Preferences and Behaviour**

Apart from the above tasks, the users are asked about the following:

1. Purpose:

a. Searching

b. Surfing

c. Development of Web Applications

d. Two of the above options

e. All of the above options

2. Usage:

a. Almost none or lesser than five minutes

b. Lesser than fifteen minutes

c. Lesser than half an hour

d. An hour or two

e. More than two hours

**4.5 Storage**

The data is stored locally in the system. The localStorage functionality of the browser is used to temporarily store the intermediate test results for a user. To store the final results of the users we use the chrome.storage.local API provided by Chrome. The permission required to access storage is “storage”.

**4.6 Classifier – Decision Trees**

Decision Trees are used for the classification process. A decision tree is an algorithm that contains only conditional control statements. It is a flowchart-like structure in which each internal node represents a test on an attribute, each branch represents the outcome of the test, and each leaf node represents a class label. In this case, the internal nodes are the tasks to be performed by the user, the branches are the outcomes of the task, and the leaf nodes represent the experience-level of the user (Expert, Intermediate, and Beginner).

**4.7 Browser Customizations**

After the user has been classified, we modify the home page of the user so that it suits his/her needs. To change this, we add the “chrome\_url\_overrides” object to the manifest. This enables us to set a page that replaces the “New Tab” page.

The home pages of different users have the following features:

1. Expert
   1. Search Engine
   2. Frequently visited sites
   3. Recommendations
2. Intermediate
   1. All features of the Expert User
   2. Links to various Browser Functions
3. Beginner
   1. All features of the Intermediate User
   2. Description of various Browser Functions

**CHAPTER V**

**RESULTS AND VALIDATION**

To understand the effectiveness of the model proposed by us, we decided to test our project using two approaches that are essential for understanding the efficacy of software based product.

(i) Mathematical accuracy of the algorithms

(ii) User Response to the product

We tried classifying users using two different algorithms- random forests and decision tress. While random forests provided an accuracy of only 72.4%, decision trees were accurate 89.8% of times. Hence, we decided to continue estimating a user’s classification based on Decision trees.

Since our project involves customization of browsers to improve user experience, we tested it with various new users. We received great feedback with respect to the ease that our customizations bring forward for users. Most users that were presented demo of the project were classified as intermediate users. 70% of these users found the shortcuts for the browser features very helpful. Most of these users also claimed that the recommendations that were provided to them were quite relevant and they are also very likely to visit top sites via the dashboard rather than the chrome access to top sites. But, the most significant part of our project was testing the browser with kids and older people who are not very comfortable with using browsers. Most of these people fell in the beginner’s category. On discussing with them, we found that almost all of these users found the customizations that were provided with us very useful.

Hence, this project has an incredible scope and can truly improve the browsing experience for multiple users by correctly combining the algorithms and features of the Intelligent Web.

**CONCLUSION**

In this project, we customized the browser for the user based on his preferences, browsing activity, and browsing skills. The first half of the project was concerned with developing a customized user dashboard that was based on the user’s browsing activity. The basic features provided by this dashboard were (i) The recommender word cloud and (ii) The easy to access top visited icons.

The second half of this project was concerned with analyzing user skills and his/her comfort with using a browser instead of his/her activities and preferences. This was done to ensure that the all kinds of users (beginners, intermediates, or experts) can have a similar kind of browsing experience. The browser was customized according to the classification results of a user to provide shortcuts and hints to users that are not very comfortable with using browsers.

Both the part of the project were tested mathematically for their accuracy and also introduced to new users for their response. The results proved to be incredible and this project has great scope in the future.

The dashboard features can be further extended to make browsing simpler for a user. The test can also be extended to capture some more user skills. While there is a good scope for extension of these features, the most significant extension can be provided to the browser customization aspect of the project. Multiple features like font size estimation, optimized search bar predictions, shortcuts and hints for browser tools can be added based on the classification of the user. Also, the project can be tested with other classification algorithms to obtain a better accuracy.

To conclude, this project successfully improved the browsing experience for a user by not only analyzing their activity but by also measuring their expertise and hence will prove to be a very beneficial solution to most people.

**REFERENCES**

[1] Ravi Kumar and Andrew Tomkins, “A Characterization of Online Browsing Behavior*”, Proc. International World Wide Web Conference Committee (IW3C2), April 2010.*

[2] Haiming Wang and Kenny Wong, “Recommendation-Assisted Personal Web*”, Proc. IEEE Ninth World Congress on Services, 2013.*