**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, 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.

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

**Summary of Results:** A taxonomy of classes was 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 consumes 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 was conducted, for a recommendation-assisted personal assistant. Data model consisted of the dataset, and recommendation algorithms. Semantic and social network analysis were applied on the Data Model layers. Latent Dirichlet Association was applied to extract topic distribution of each document, and community mining algorithms to cluster nodes by their links. Recommendation algorithms were formulated accordingly, and weighted. The ﬁnal user selection will be stored in the query history of the user model, making the system adaptive.

**Summary of Results:** It was 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 was built, and history API was used to extract browser history of a user. The history dataset collected was then cleaned by removing all query strings from the URLs, while only maintaining the base URL. The frequency of each URL was 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, was 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 were taken to be used as icons for the dashboard items, however, this approach required 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 was required. A list of 100 most popularly used sites was obtained, and short textual descriptions for these sites was scraped from the Internet to build a base dataset for the similarity algorithm. Unigram models were constructed from the descriptions, and stop words were eliminated. Similarity between each pair of URLs was calculated, using a cosine similarity formula for the significant words in the description of the website as found. An upper triangular similarity measures matrix was constructed, using the cosine similarity formula. Since the generated matrix was too large to use in the program, a list of ten websites with highest similarity scores was maintained for each website. These top ten websites were recommended to a user whenever some website was prioritized enough by the formula used to rank website preference from the user's history.

**3.4 Recommendation Visualization**

A Word Cloud was used to visualize the results of the recommendation algorithm. Word Cloud was chosen to represent the results, because considering the user-friendly intention of the application, it was the best visualization tool for the purpose. Sites which were highly recommended were highlighted in the Word Cloud by displaying them in larger font sizes, and sites with lesser recommendation were 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 aethetic 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 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 decided to classify users based on their expertise level by providing them with a chrome extension based test. This test was 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 was collected by creating a quiz which tests the familiarity of the user with the functionality of the browser. The data was 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 was 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 were 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 was stored locally in the system. The localStorage functionality of the browser was 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 were 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, 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**

**CONCLUSION**

**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.*

GUIDELINES FOR THE PREPARATION OF SEMINAR REPORT

1. Seminar report should be typed neatly only one side of the paper with single line spacing preferably on a A4 size bond paper (210\*297). The margins should be : Left

– 1.25. Right –0.75, Top and Bottom –0.75

2. The total number of reports to be prepared are

One copy to the department

3. Before taking the final print out, the approval of the concerned guide(s) is mandatory and suggested corrections, if any, must be incorporated.

4. Organization of the report should be as follows

* + 1. Inner title page
    2. Certificate
    3. Acknowledgement
    4. Abstract
    5. Table of contents
    6. List of tables and figures

5. Table of content should contain the following topics.

* 1. Introduction
  2. Literature survey
  3. Headings ( in the paper)
  4. Results discussion
  5. Conclusion
  6. References

6. Chapters must contain introduction which usually specifies the scope of work and its importance and relation to previous work and present development. Main body of the report divided approximately into chapters. Sections and subsections

7. Chapter sections and subsections may be numbered in the decimal form e.g. Chapter 2, sections as 2.1, 2.2 etc and subsections as 2.2.1, 2.2.1 etc.

1. The chapter must be left margin justified (preferably use font size 16). Followed by the title of chapter usually centered (Preferably use font size 18), section/subsection numbers along with their headings must be left justified with section number and its heading in font size 16 and subsection and its heading in font size 14. The body or the text of the report should have font size 12.
2. The figures and tables must be numbered chapter wise for e.g. Fig 2.1(here 2 is the Chapter no) ,Table 3.2
3. The Conclusion chapter should contain the summary of the work carried, contribution of any, their utility along with the scope for further work.(Conclusion)
4. References: The reference should be numbered serially in the order or their occurrence in the text and their numbers should be indicated within square brackets for eg[3].
5. Seminar report Outer cover color should be gray. And plastic cover should be used(**soft bound**).
6. Font should be Times New Roman
7. Print out: Black

Report should be minimum of 15 and a maximum of 20 pages (single sided).