CASE-BASED REASONING FOR AN ADAPTIVE WEB USER INTERFACE

Farhi Marir
Knowledge Management Research Centre, School of Computing,
London Metropolitan University
F. Marir@londonmet.ac.uk

ABSTRACT

This paper presents the design and development of a widget (CBaUI) which resides on top of a web user interface and which uses case-based reasoning paradigm to model the user profile and adapt the content of the host web site. It uses click and mouse movement heat map techniques to track and record the user browsing behavior which will be analyzed by text mining algorithm to form browsing patterns. Then nearest neighbour algorithms analyses and index the user browsing patterns into groups of web user profiles and stores them into a case base memory. The user profiles cases will be retrieved and used to adapt and personalise services of the host web site to match user profile. It also provides facilities like content and collaborative search facilities useful for each web user profile, chatting facilities for the user to seek support and guidance from the admin of the host web site and display of the minimum content and services of the host web site that is most appropriate to that web user profile. CBaUI was tested and the results have shown improvements of host web sites.

KEYWORDS
Machine Learning, Nearest Neighbour Algorithms, Case-Based Reasoning, Web User Modelling; Adaptive user interface; Usability and learnability

1 INTRODUCTION

Adaptation is referred to the notion of changing something to meet some specific requirements or purposes [1]. Adaptive systems are described as tools which develop new information about how to do the task better by analysing past experience and relating it to performance criteria set by humans [2]. It is also stated that an adaptive system adapts its behaviour to individual users based on information about them which is either implicitly collected during the user-system interaction or users are explicitly asked for it, and the adaptive system performs the adaptation using some form of learning, inference or decision making [3].

While it is recognised that adaptive interfaces improve usability, users’ experiences and Learnability and bring potential gains and cost-benefit trade-off for usability, critics argues that autonomous user interface adaptation may disorient users and reduce its usability. Most critics relate these limitations to adaptive user interfaces unpredictability and lack of accuracy [4], [5] and 6.

In this research adapted user interface is viewed from usability and learnability point of views. Two main approaches have been undertaken to develop such adaptive user interfaces to address the reported unpredictability and lack of accuracy limitations: (i) those adaptive (static) user interfaces which are designed based on usability, learnability methods and user feedback are used at the design level only and (ii) those adaptive (dynamic) user interface which are also designed similarly but they are
different in that they keep continually learning web user profile to dynamically adapt themselves to the needs of that existing or discovered new web user profile.

A number of methods have been used for the design of static adaptive user interfaces. For instance, analytical usability and learnability studies which involve analysis of the user interface using inspection methods such as heuristic evaluation, cognitive walkthroughs, GOMS analysis and so on [7]. Also, empirical usability and learnability methods which involve people using usability and learnability test methods and traditionally conducted as either formative or summative studies [8]. Formative studies are carried out during the product development process with the aim of fixing problems found during the product development process whereas summative studies are carried out after completion of the development and are used more as a basis of deciding lessons learned and base lining a product [9]. The activities undertaken here can include recorded/measured usability and learnability evaluation, interviews, and experiments and so on.

However, dynamic adaptive user interfaces are enhanced with learning user profile which could be either Informative interfaces that focus on filtering information the user finds interesting or useful or Generative interfaces that generate some useful knowledge structure to support the user in their experience with the user interface [10].

In this paper we present a dynamic case-based adaptive user interface (CBaUI) widget which is portable as it can be hosted on top of any web user interface by adding few line of code to the header of the host web site. It uses case-based reasoning paradigm and machine learning algorithms to learn web user profile while browsing through the web site to enhance the usability and learnability of the host web site and adapts it to respond to that profile needs. CBaUI widget is an extension of a JISC project commissioned to develop an adaptive interface to improve usability and learnability of NaCTeM e-Research tool (http://www.nactem.ac.uk/).

The paper is divided into five sections. Section two will be devoted to previous work on developing static and dynamic adaptive user interfaces, section 3 devoted to both usability and learnability methods used to design CBaUI widget and also the development of its learning of web user profiles, section four will be presenting and evaluating the usability and learnability features of CBaUI and finally section five will be devoted to conclusion and future work.

2 PREVIOUS WORKS

Several recent works have been sponsored by JISC to develop usable and learnable user interface\(^1\). Most of them could be considered as static adaptive interface as usability and learnability are dealt with at the design level only. It includes, ALUIAR project ranks the results of a user group feedback through interviews and \textit{“walk through”} to improve the usability and learnability of the user interface of Synote, the open source web based video and audio annotation tool.

Rave in Context project developed usable, accessible, learnable and adaptable W3C widget templates and widgets for MyExperiment, Simal and OpenDOAR web sites.

USeD project developed a new user interface to improve the usability of Digimap data downloader, making it

\(1\) http://www.jisc.ac.uk/whatwedo/programmes/researchinfrastructure.aspx
easier for a range of subscribers to use. It was designed by creating a number of stereotypical user Personas which are based on the actual Digimap user requirements, their expectations of the service and their knowledge of spatial data.

ReScript Usability/Learnability Enhancement project improved usability and learnability of the user interface of ReScript, a prototype of digital editing and research environment originally developed to support collaborative work on historical texts. The feedback compiled from a series of online surveys and interviews with editors and researchers were used to make change to ReScript interface to meet the needs of a variety of researchers working with very different texts, and with differing levels of expertise.

The Word Tree Corpus Interface had the goal of providing an alternative interactive user interface to traditional text-analytical tools like KeyWords In Context (KWICs). The website produced allows users to generate word trees for individual terms, starting from the searched term at the leftmost edge with branches of proceeding words extending to the right. To develop a usable and learnable user interface, the Word Tree Corpus team undertook a combination of quantitative and qualitative studies through surveys and video interviews of stakeholders. They also used Google analytics to analyse user behaviour and to further improve their new user interface to reflect the users' needs.

Also several works have been undertaken to develop dynamic adaptive interfaces. Reference [11] reported the development of SySKILL & WEBERT adaptive web user interface, which recommends web pages on a given topic that the user is likely to find interesting. The user marks suggested pages as desirable and undesirable and the system uses naive Bayesian classifier for this task, and demonstrates that it can incrementally learn profiles from user feedback on the interestingness of Web sites. Furthermore, the Bayesian classifier may easily be extended to revise user provided profiles.

Another web user interface NEWSWEEDER system [12] recommends stories to the user using each word in the story to predict whether the user finds the story interesting or not.

P-TIMS [13] is a commercial financial management system was revised to add an adaptive and adaptable interface using a simple user model and rule set. As the user spends more time using the system and uses more complex functions, the system reveals a more extensive interface. The user model is explicitly exposed by providing a "preferences" dialog box, which the user can adjust at any time.

AVANTI [14] is a hypermedia information system about a metropolitan area uses an initial interview to create the initial primary assumptions (i.e., user profile), draws inferences to generate additional assumptions, and uses stereotypes for certain subgroups of users (e.g. tourists, blind users). It then customizes the web pages presented to the user accordingly.

Interbook [15] is an adaptive system which derives much of their data for the user model from the user interface component, which can track user actions and report them in detail to the application user model. Interbook addresses these problems by tracking what the users have seen, rather than what they have done, and using that to infer what the users know.

Most recently, Reference [16] developed a user interface prototype for the Android smartphone, which recommends a number of applications to best match the user's context based on five variables; time, location, weather,
emotion, and activities. The developed system derives the best three recommended applications based on a probabilistic learning and inference algorithm named “Spatiotemporal Structure Learning” which extends Naive Bayesian Classifier.

It can be noticed that most of the above dynamic adaptive interfaces require explicit input from the user to be able to model web user profile and adapt web user interfaces to satisfy individual user profiles. CBaUI which is also a dynamic adaptive web user interface implements an intelligent and portable widget, which provides both explicit and implicit input from the user to build web user profiles. In the implicit input, CBaUI uses automatic tracking mechanisms to track and record the user behaviour while browsing the web site on which CBaUI widget is hosted also uses inductive learning algorithms to identify or discover new web user profiles and enhances the host web site usability and learnability capabilities accordingly. Furthermore, it detects changes of hosted web site content in terms of functionalities and resources while tracking the user browsing and reflecting that in the adaption of the web user interface and also the services presented to the user.

3 DEVELOPMENT OF CBaUI INTERFACE

3.1 CBaUI Web User Interface Design

In CBaUI project, adaptive web user interface is seen as an interactive software system that improves its usability and learnability in its interaction with a user based on partial experience with that user. The formal definition of usability by the International Standards Organisation\(^2\) is that usability is concerned with the “effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments.” Effectiveness means how accurately and completely users can achieve their goal. Efficiency means the effort required to achieve a goal. Satisfaction means the comfort and acceptable of using the system to achieve the goal. These definitions are in keeping with [7] who also stressed the importance of learnability which could also be considered an important aspect of usability.

According to the Usability First glossary and Reference [17] learnability is a measure of the degree to which a user interface design can be learned quickly and effectively. Learning time is the typical measure. User interface designs are usually easier to learn when they are familiar and designed to be easy to use based on core psychological properties. Through literature reviews the learnability of a user interface design can be broken down into five: Familiarity, Consistency, Generalizability, Predictability, and Simplicity. It is also stressed that although learnability could be part of usability, little is shown that an increase of ease of use (usability) can be realised without actually improving the user’s mental model (learnability) of adaptive systems.

Several methods have been used for the design of usable and learnable user interfaces. Analytical usability studies involve analysis of the system using inspection methods such as heuristic evaluation, cognitive walkthroughs, GOMS analysis and so on [7]. Empirical usability involves people using test methods and traditionally conducted as

\(^2\) http://www.usabilitynet.org/tools/r_international.htm
either formative or summative studies [8]. Formative studies are carried out during the product development process with the aim of fixing problems found during the product development process whereas summative studies are carried out after completion of the development and are used more as a basis of deciding lessons learned and base lining a product [7]. The activities undertaken here can include recorded/measured usability evaluation, interviews, and experiments and so on. Reference [18] research on “estimating the relative usability of two interfaces” concluded that the most reliable way of determining the relative performance was through the use of empirical usability studies rather than analytical usability studies although the empirical studies are more expensive to perform.

CBaUI was designed based on a methodology commonly found in usability studies and recommended by Reference [7]. It uses personas and scenarios as a way of contextualising the what, where, how, when and why of the use of an application so that in essence it shows who the application is targeted to. Power of personas is that all stakeholder involved in the design process are much more likely to engage with other people, real or fictional, than they are with statistical information [19].

We conducted an experiment within the university research community on the impact of usability methodology of personas and scenarios that led to the design of the CBaUI interface as a widget that can be hosted on to of a web site as shown in “Fig. 1” on the left hand side of NaCTeM web interface.

Figure 1: CBaUI Widget hosted by NaCTeM
As shown in Figure 1, CBaUI can be hosted on any web site by adding few lines of codes to the host website headers files. It provides adaptive web user interface widget is comprised of a free-text / autocomplete search box which offers enhanced search capabilities for host website. The learners can also specify explicitly their profile e.g. skill level for NaCTeM host web site, and can additionally choose from a variety of links which are suggested according to their user level. At the bottom of the widget, a ‘live help’ box is provided to enables users browsing the host website to communicate with a website administrator in real-time.

3.2 Modelling Web User Profile
CBaUI project used case-based reasoning (CBR) paradigm [20] and [21] to learn web user profile and develop the CBaUI adaptive web user interface. CBR uses machine algorithms solves new problems by adapting solutions of previous similar problems following the cycle Retrieve, Reuse, Revise and Retain as shown in Figure 2. Like for instance CBR systems CLAVIER[22] which are developed as advisory systems recommends loads and layout for aircrafts parts to be cured in an autoclave and CBRefurb [23] which retrieves previous similar refurbished building to estimate the cost of new refurbishment. Both use inductive learning algorithms within CBR to index and retrieve and
revise/adapt most similar past cases from their case libraries.

To build a case base you need to design a case representation, indexing scheme and retrieval system.

### 3.2.1 CBaUI Case Representation & Indexing

CBaUI use Click / mouse movement heat maps techniques to tracks the user interaction with the host website as shown in Figure 3.

The results of click heat maps tracking are translated by CBaUI into text format and stored as a personal user record in CBaUI database. Then the CBaUI text mining tool analyses the collected user text records to derive user patterns (clicked key words, functions, web site links, etc…) which will be represented as new case (vector) of the current user updating the case base. The indexes of the case will be represented by the keywords and web pages browsed by the user. The new case will be further analysed by the CBaUI case-based inductive algorithm to determine whether it is a new or an existing web user profile and therefore updating if necessary the content of the category structure in Figure 5.

### 3.2.3 CBaUI Machine learning Algorithm

Given a description of a problem, a retrieval algorithm, using the indices in the CBaUI case-memory, should retrieve the most similar cases to the current problem or situation. The retrieval algorithm relies on the indices and the organisation of the memory to direct the search to potentially useful cases.

The issue of choosing the best matching case has been addressed by research into analogy [24]. This approach involves using heuristics to constrain and direct the search.

Case-based reasoning will be ready for large scale problems only when retrieval algorithms are efficient at handling thousands of cases. Unlike database searches that target a specific value in a record, retrieval of cases from the case-base must be equipped with heuristics that perform partial matches, since in general there is no existing case that exactly matches the new case.

In order to retrieve or generate suggestions on the web user profile which are relevant to the given browsing pattern we had to consider amongst well-known methods for case retrieval like nearest neighbour, induction, knowledge guided induction and template retrieval. We found that the most effective algorithm to match users with results according for instance to their skill level in NaCTeM was Nearest Neighbour Algorithm (NNA) because it is more effective when the case base is not huge.

NNA involves the assessment of similarity between stored cases and the new input case, based on matching a weighted sum of features. A typical formula used in nearest neighbour
algorithm is explained in [25] and shown below in (1). In the formula, $W$ is the importance weighting of a feature (key word, web link, etc...), $\text{Sim}$ is the similarity function, and $f_I$ and $f_R$ are the values for feature $i$ in the input and retrieved cases respectively.

$$\frac{\sum_{i=1}^{n} W_i \cdot \text{Sim}(f_I^i, f_R^i)}{\sum_{i=1}^{n} W_i}$$

(1) Formula for Nearest Neighbour Algorithm

Furthermore, Induction algorithms ID3 [26] which determine dominant is used to discriminate cases based on these features, and generate a decision tree type structure to organise and categorise CBaUI cases of web user profiles in memory.

4 CBaUI ADAPTIVR FEATURES

In addition to its design using an iterative design methodology that focuses on usability, learnability, participatory design suitable for service-based implementations, CBaUI widget used case-based reasoning and inductive learning methods to carry out learning about both the user profile and the host web user interfaces to promote usability and learnability and provide the right and useful information the user needs.

CBaUI has addressed most of the common issues reported in the literature that arise in developing adaptive or advisory interfaces including information filtering, supporting the user in his experience with the web site and also make physical changes to user interface itself.

4.1 CBaUI Filtering Information

CBaUI has implements advanced filtering algorithm(s) for extracting digital content of the host web site like NaCTeM to match user needs. Content and collaborative based filtering methods [27, 23] have been used as the basis for selection and learning about the content of NaCTeM web site. Content methods suggest topics similar to ones a user group with similar profile has liked in the past.

However, collaborative filtering methods suggest items outside the user’s normal area that the user will still find interesting, as the basis for selection and learning.

4.2 CBaUI Web User Interface Adaptations

Also, based on the identified web user profile, CBaUI recommends choices on some aspects of the research processes as desirable or undesirable, rating them on a scale and giving some similar form of evaluation to help the user in his selection of retrieved information. An instance of CBaUI user interface adapting itself to a category of users is shown in Figure 4 below. In this screen shot it suggests few links in yellow colour usually used by users with similar web users profile to the user currently browsing the host web site.

![Figure 4: Suggesting useful links for a particular category of users](image)

4.3 CBaUI Live User Support

As shown in Figure 5 below, CBaUI interface also provides live communication facilities through ‘live help’ box enabling the user to seek support and guidance from the NaCTeM
In a recent study, the investigation undertaken in [28] found that users lose up to 40% of their time due to "frustrating experiences" with software’s user interfaces, with one of the most common causes of these frustrations being missing, hard to find, and unusable features of the software.

In order to assess the frustration of the users and evaluate the CBaUI widget based on some usability and learnability criteria, we invited around fifty research and students with different skills to run NaCTeM web user interface with and without CBaUI widget. The measure we used for evaluating CBaUI widget usability and learnability features were based on completion of tasks on time, accuracy in providing information to the user and also predictability of user interface.

They were asked to complete a series of simple tasks using the widget. Different researchers were asked to complete a series of tasks without the widget. We compared the results of the two groups to get an idea of how effective CBaUI is at enhancing the experience of users.

The results of this evaluation have shown that all users who used NaCTeM with CBaUI widget completed all the tasks on time compared to 80% when using NaCTeM with CBaUI widget.

We also decided to know what are the actions predicted by CBaUI widget compared to what the user intended to do. How many of these actions are actually are accurate. For this we conducted a second evaluation test with 45 researchers and students to assess predictability and accuracy of CBaUI.

Results on predictability and accuracy test has shown that 75% reported that the CBaUI widget predicted what they are looking for and 85% reported that CBaUI provided accurate information they asked for. This is compared to around 15% predictability and 75% accuracy when using NaCTeM interface without CBaUI respectively.

5 CONCLUSIONS

In this paper we presented the design and development of an adaptive web user interface in the form of an intelligent and portable widget named CBaUI that could be hosted on top of any web site user interface to enhance its usability and learnability features.

CBaUI combines click heat map techniques, with text mining and case-based reasoning with nearest neighbour algorithms to model the user profile and adapt accordingly the web user interface to that user profile. Furthermore, CBaUI provides chat facilities to support the user in browsing efficiently the web interface and it implements advanced information retrieval techniques like content methods suggest topics similar to ones a user group with similar profile has liked in the past and collaborative filtering methods suggest items outside the user’s normal area that the user will still find interesting.

CBaUI has been evaluated with NaCTeM e-Research web interface and showed the improvement of NaCTeM interface in terms of usability and learnability of the users.

6. ACKNOWLEDGEMENT
The Author acknowledge the JISC funding for sponsoring this research work and also Dr. S.P. Sahithi and Dr. Y. Jing who contributed to the initial eRaUI JISC funded Project.

6 REFERENCES


