Non-Invasive Browser Based User Modeling Towards Semantically Enhanced Personlization of the Open Web

Kevin Koidl, Owen Conlan, Lai Wei, Ann Marie Saxton
Knowledge and Data Engineering Group
SCSS, Trinity College Dublin
Dublin, Ireland
{kevin.koidl, owen.conlan}@cs.tcd.ie, {weil, sextonam}@tcd.ie

Abstract—Currently the user’s web search is disjoint from the resources which is subsequently browsed. Specifically the related instances of the search are not displayed on the following pages. This lack of continuity between the actual search and the web sites displayed may lead to skimming by the user to identify what is relevant on the pages. This paper presents an approach to the continuous modeling of a user’s interests through a browser based plug-in that may be used to inform the intelligent highlighting of content on the browsed pages. Specifically the paper presents two related systems; one for capturing and modeling the user’s browsing behavior and the second for leveraging the power of linked data to highlight information relevant to the user’s interests. The design, implementation and evaluation of both systems are presented as well as a conclusion and outlook for further work.

Keywords—web personlization; implicit user modeling; web-based content management systems; linked data

I. INTRODUCTION

The usage of the web is heavily influenced by the need to find relevant information. Within the vast range of mechanisms to assist this task the use of keyword driven search engines is clearly the most dominant application. However search engines cannot assist the user once the desired page is displayed resulting in a disjoint between the users search and the web page the user chooses to browse to. This gap can lead to a browsing behavior in which the user has to manually identify the relevant sections on the page. Specifically this information can be used to deliver a personalized experience to the user by highlighting related keywords or sections within the web page.

The first project Automated Modeling System (AMS) concentrates on the identification of the user’s interest. The main goal of AMS is to identify the key areas of interest when browsing information and to do this in a non-invasive manner. For this the system captures, analyses and utilizes information in order to create a unique individual user model and provide a summarized view of the user’s interests. Furthermore AMS is a implicit user modeling service that works in the background by modeling the user’s browsing behavior without the user being prompted to provide further information.

The second system Semantically Enhanced Open Wiki (SEOW) delivers a personalized browsing experience by highlighting relevant keywords or sections within a wiki page related to the interest of the user. As mentioned above the use of the linked data initiative allow a broader and more valuable feedback by highlighting higher level concepts related to the initial interest. Similar to AMS the non-intrusiveness and user centric design is a main design goal. The user can adjust the highlighting depending on which level of knowledge is relevant i.e. if only the specific plane type or higher level concepts such as all plane types are to be highlighted. Furthermore SEOW strictly separates content, knowledge and intelligence by implementing a MediaWiki extension interfacing with linked data as well as a rule based engine to infer decision related to the highlighting of relevant text.

The remainder of the paper is structured as following. First a browsing scenario is introduced to motivate and illustrate the usage of both systems. This is followed by an illustration of the architecture and design of both implementations. Finally the findings of the user based evaluation are discussed, followed by a conclusion and outlook to further work.

II. BROWSING SCENARIO

The following browsing scenario has the goal to illustrate the usage of both systems to enrich the browsing experience of the user based on the initial interest and related information retrieved from linked data.

We assume that John is interested in the Concord aircraft. Furthermore we assume that John is also interested in any related information. For this John browses across various
different web pages related to information about the Concorde. In this scenario we assume that John starts to follow his interest by using a key word based search engine. In this he states the keyword ‘Concord’ prompting a result list. John selects various web pages from the result list and browses these. After several pages he decides to browse to the related Wikipedia page (note Step 1 in figure 1). John’s previous web page visits were assessed by the Automated Modeling Service (AMS) creating a user model which reflects John’s browsing interest. John can view his user model at any time by clicking on the plug-in item in the upper corner of the browser. The resulting view indicates key terms related to John’s main browsing interest as well as an indication which sites are currently most visited (note Step 2 in figure 1). If John wishes more scrutiny in relation to his user model John can choose to view all database entries by accessing the Google Chrome extensions option. After reading the Wikipedia page related to Concorde John starts building up interest in other related sites, such as what Airlines were flying the Concorde and from which Airport did it depart the most. For this John starts to browse to related Wikipedia and external web pages within and from Wikipedia for example about British Airways and Heathrow. John now decides to use the information collected in AMS to provide him with additional knowledge related to his interest. For this John uses the Semantically Enhanced Open Wiki (SEOW) system. SEOW retrieves related concepts from linked data provided by dbpedia. Instances of these concepts are then highlighted in the text. To avoid information or cognitive overload John can select the different concept related to his specific interests in a new window in the upper right corner of the wiki page (Note Step 3 in figure 1). Based on John’s selection the relevant instances of the selected concept are highlighted in a predefined color scheme. In this example all instances of Airports such as London Heathrow or London Gatwick Airport are highlighted brown and all instances of airline alliance such as OneWorld are highlighted in pink (note Step 4 in figure 1). John is now able to select relevant concepts related to his previous browsing interest leading to a more effective and richer browsing experience.

III. ARCHITECTURE, DESIGN AND IMPLEMENTATION

In this chapter an overview of the architecture and design of both systems will be provided. The main design difference between both systems is that AMS is purely browser based and SEOW is server sided. This allows a clear separation of concerns and ensures that sensitive information such as the users browsing interest and most frequent site visits remain on the client side therefore under the user’s control. Furthermore the server side design of SEOW allows a clear separation of concerns such as ensuring that all personalization features are performed at the source of the information. This ensures that the personalization features are browser independent. The following introduces both systems separately and indicates overlapping functionalities.

A. AMS

AMS is the implicit user modeling service to identify the user’s browsing interest both non-invasive and user controlled. Based on the easy to use developer extension AMS was implemented as Google Chrome plug-in. Following design requirements were identified: (i) text analytics, (ii) the rating methodology, (iii) data storage and finally (iv) the user interface. In the following all four design requirements are introduced followed by a brief overview of implementation details.

In order to generate usable data from the information collected text analytic mechanisms were applied. These mechanisms include stop word filtering and the use of porter’s stemmer algorithm [1]. This was followed by a word frequency count. The result is then multiplied by a separately calculated rating value. This rating value is based on two indicators. First, the time spent on a page. This is necessary in order to avoid that key terms which are only viewed for a short time are not rated the same as ones which are viewed longer. Furthermore the mouse movement is used as indicator idle time in which the user may not be actively reading the displayed page. Once the user exits the web page the idle time is subtracted from the total viewing time. The second indicator is the length of the page. The goal of this is to avoid an unfair balance of short pages in relation to long pages. For this the available screen resolution is taken into account. Finally the resulting viewing time is divided by the resulting page length. The result of this calculation is defined as ‘time spent on page’ which is multiplied with the term frequency resulting in the final score per keyword.

The third requirement is related to the data storage. To address privacy concerns all information’s are stored locally in the web browser. For this HTML5 based SQL database is used and managed via the Google data store extension. Within the SQL data store two tables are hosted. The first table stores keywords and their corresponding frequency, the second table stores the visited web pages and their corresponding visit frequency.

The final design requirement is related to the user interface. Based on the non-intrusive nature of the approach the user has to click on a provided AMS icon on the upper

2 http://dev.w3.org/html5/spec/Overview.html
right corner of the browser window to view the user model related information (note Figure 1 Step 2). For a high level illustration of the AMS architecture reflecting the design requirements please note Figure 2.

![High level architecture of AMS](image)

As mentioned above AMS was implemented as a Google Chrome Browser plug-in in JavaScript. The web page is processed based on the already illustrated requirements such as different text analytics methods. In order to process the web page efficiently it is parsed via the Document Object Model (DOM) functions. Once the web page is processed successfully the final result is written back to the data storage within the Google Chrome Browser. The user can now view the result by displaying the AMS browser window indicating the top key terms identified as well as the top three visited web sites. In order to enrich the user’s browsing experience AMS can query the linked data such as dbpedia allowing the retrieval of higher level concepts. Furthermore AMS allows API functionalities to access user model data via third party server side services such as SEOW which is introduced in the following.

B. SEOW

The main design difference between SEOW and AMS was the decision to implement SEOW purely on the server side. The main reason for this was the clear separation of concerns and to ensure browser independence. In the following the three separate concerns are introduced including the related design requirements:

(a) The use of a web-based Content Management System (WCMS) to manage the creation and editing of the web content; (b) A Knowledge component to find the type or class of the identified resource; (c) An intelligence component to store, manage and execute adaptation rules. This separation of concerns resulted in the implementation of three individual by interlinked sub-systems.

The content component (a) was implemented as a MediaWiki extension defined as SemanticTag extension. This extension interfaces with the other two components of the SEOW. It is implemented in PHP and strictly follows the MediaWiki extension guide lines. The Semantic Tag extension is both the applier of the adaptation and the communication system of the content component. The extension adds a new tag into the wiki text defined as `<semanticTag>` which is used as hook for a PHP function call. This function manages the request to and answers from the adaptation web service Intelligence component. The Semantic Tag extension is also responsible for adding text elements such as the highlighting and the selection window.

The knowledge repository (b) is implemented as a REST web service in Java for seamless integration with different web application servers. The role of the knowledge repository is the investigate linked data properties such as higher level concepts via dbpedia. This information is then used by the intelligence component. The intelligence component (c) is implemented as a REST web service in Java. This component manages the adaptive rules which are defined in XML. All messages between the sub-systems are encoded in JSON based on its support for both Java and PHP. The intelligence component relates the content source to the user’s interest by mapping algorithms encoded as adaptive rules. These rules can be extended and identified via third party information such as AMS.

In the following the adaptation rules and the adaptation process is described. A SEOW adaptation rule entails a list of DBpedia URIs of object types that are relevant to a certain type of objects or topics, and it is stored in XML format. Therefore there is a separate rule for objects of different types. In order to create the list, a domain expert who knows exactly what the most relevant object types are is needed to add the corresponding URIs of the types from DBpedia.

![Sample adaptation rule](image)

The adaptation process triggering an adaptive rule is: (1) resolve the page topic, (2) load the adaptation rules, (3) construct the adaptation results and finally (4) return the adaptation results to the content repository.

---

3 http://www.mediawiki.org/wiki/MediaWiki
The evaluation of both systems was conducted with two different groups with each group evaluating one system to allow a more selective and precise evaluation.

A. Evaluation and key findings of AMS

The key to the success of building an accurate user modeling service is the satisfaction of the user and most importantly to what degree their interests have been inferred. The purpose of this evaluation was to evaluate the functionality, usability and the overall appeal of the implemented Google Chrome extension as well as to investigate the potential benefits of utilizing linked data to return more relevant results to the user. Six users were chosen to perform the main evaluation of the system and three of them were asked to choose a domain to work within. The three domains chosen were Life Coaching, Sport and Fashion. The remaining three participants worked in the open domain. The group consisted of two male and four females from a variety of backgrounds. Four of the users fall in the 35-45 age bracket, one in the 18-25 bracket and one in the over 45 bracket. Users were asked to simulate their normal web browsing behavior. To illustrate the added value linked data concepts related to the indicated interests were presented. The evaluation was conducted in three phases. First a pre-evaluation questionnaire to investigate the user’s understanding of the web and the use of user modeling; this was followed by a 30 minute browsing session and finally the results of this browsing session were displayed to the user by the AMS plug-in browser window. In a post evaluation questionnaire these results were then evaluated including showing user related information from linked data.

The pre-questionnaire indicated three key findings: First, that the majority of users spend between 11 and 15 hours a week searching or browsing for information, which accounts for more than half of their total browsing time. Second, none of the users like to fill out profile building questionnaires while browsing the internet which gives merit to the noninvasive approach of AMS. And finally half of the users have difficulty in finding what they are looking for when searching the internet. All users agreed that the websites and their ratings as shown by AMS were symptomatic of the websites that they found most interesting during the period of the study. Related to the additional information provided to the user based on linked data the users perceived these mostly as useful and relevant. It has to be noted that there was clear difference between the users that conducted their evaluation using a chosen domain and those who used the open domain. The users in their chosen domain all strongly agreed that the information shown to them from the linked data environment was useful and relevant to what they were looking for. We can go as far as to say that this part of the evaluation generated some excitement from the users as they had never seen or heard of linked data before and they were pleasantly surprised by the way that all the relevant information was there without everything else that they are used to seeing from regular searching.

The evaluation of AMS was successful insofar as all users agreed that the websites and keywords shown by AMS were indicative of their interests while browsing the web. Users were in favor of the idea of a user model working incognito, building a profile automatically as opposed to having to answer questions which all users found annoying. Most users felt that an accurate user model could ultimately save time and return more relevant information.

B. Evaluation and key findings of SEOW

The main goal of the SEOW evaluation was to investigate the usability of a third party adaptation service to adapt content on open data. Moreover the completeness and accuracy of the adaptation performed was investigated.

To evaluate of SEOW, a total of 52 distinct pages about civil aviation topics, including airlines, airports, airplanes, and their manufacturers, were created in MediaWiki. The content of the pages were copied from their corresponding Wikipedia pages with slight modifications to hide irrelevant content and links. After this six tasks divided into three groups were prepared. Figure 4 shows the designed tasks.

<table>
<thead>
<tr>
<th>ID</th>
<th>Group</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>I</td>
<td>List all airlines that ever operated Concord.</td>
</tr>
<tr>
<td>B</td>
<td>I</td>
<td>Find the airline that received the delivery of the first Boeing 717 aircraft.</td>
</tr>
<tr>
<td>C</td>
<td>II</td>
<td>Find all hub airports of British Airways, and for each airport, find other airlines that also use it as a hub.</td>
</tr>
<tr>
<td>D</td>
<td>II</td>
<td>List the founding members of the airline alliance that Lufthansa is in.</td>
</tr>
<tr>
<td>E</td>
<td>III</td>
<td>Among the airlines that belong to the parent company of City-Jet, find the one with the biggest fleet size.</td>
</tr>
<tr>
<td>F</td>
<td>III</td>
<td>Among all the airlines that use Dublin Airport as hub, find the one with the smallest fleet size.</td>
</tr>
</tbody>
</table>

Figure 4. Tasks in SEOW user tests

The evaluation was conducted with a group of eight volunteers, five male and three female, with mixed Wikipedia experience and knowledge levels in civil aviation. After introducing the functionality of SEOW to the participants the nature of the tasks were instructed. They were then asked to finish the six tasks in a particular order. For the first three tasks, which included one task from each task group, users were given a clean MediaWiki instance without the adaptation service; for the last three tasks, which also contained one task from each group, the highlighting options were provided and the users were encouraged to use them to assist in the task. Prior to the browsing tasks the participants were asked to fill in a pre-questionnaire to assess their familiarity of MediaWiki and the topic of civil aviation. After finishing each of the last three tasks, the participants were asked questions related to the degree in which the adaptation service helped them in fulfilling the tasks. This questionnaire also included System Usability Scale (SUS) questions to evaluate the usability of the adaptation service. During the evaluation a screen video capture program – Camtasia Studio version 7.0.111 - was used to record the users' interactions with system. Information such as time to answer and steps taken were gathered from analyzing the recordings. Based on the analysis of both the participants' evaluation results and the system's performance, the following key findings can be concluded:

Over 75 percent of the participants stated that SEOW was helpful in finishing the tasks. Moreover even with limited
content, very simple adaptation rules, and partly incomplete linked data knowledge, the adaptation service in SEOW was undoubtedly perceived as usable and useful. In relation to the accuracy rate of the task answers an increase of 16 percent was indentified leaning towards the assumption that SEOW helped the user in finding more relevant or more precise information possibly overlooked without the highlighting mechanism. However the user did not spend more time on the text which would have explained the higher answer accuracy rate; the overall time to answer even decreased by using the system.

Overall the participants mostly agreed that there is added value in using SEOW on open data such as wiki text.

V. RELATED WORK

User modeling systems are extensively studied in the application area of Adaptive Hypermedia (AH) [2]. For adaptive systems it is essential to develop a clear understanding of the user’s needs and preferences in order to provide useful personalization. However most AH mechanisms cannot be applied to open data, based on the divers nature of document models and user interests [3]. In the following different approaches to overcome this shortcoming are discussed.

Several user modeling service have been introduced in the past. The User Modeling Tool (UMT), as one example, proposed by Brajnik et al. [4] specifies stereotypes, containing user type descriptions in the form of attribute-value pairs. The stereotypes are arranged in random hierarchies and sub-stereotypes can inherit information from the main stereotypes. Every stereotype has triggers and these can indicate when a stereotype is applicable to the current user. A rule interpreter provided by UMT allows the defining of user model rules. Assertions about the user are stored by UMT, and are generated by the application system. In recent year’s user modeling services has shifted towards ontology based user modeling. Kay et.al. [5] outlines the advantage of using ontology’s in user’s modeling and states that a user model "[…] needs an agreed ontology and representation so it can be used by different application programs". The idea of a shared user model is introduced with the user modeling system Personlis [6]. In this a server side approach is introduced, which bases its user model on component-evidence-source triplets. Each application can define its own triplets without having to regard the others, which limits its reusability. OntoBUM (Ontology based User Model) [7], as another example, is an ontology-based user modeling framework which was developed mainly for knowledge management systems. In OntoBUM there are two parts to the user model, an explicit and an implicit part. The implicit part is concerned with system usage where the authors characterise users as ‘readers, writers or lurkers’, while the explicit part holds qualities such as the profile, preferences and identity information. Letizia [8] as a third example scans the WWW ahead of the user, investigating links which are on the current webpage, and in turn using this information for the user model to recommend pages that it has determined will be of interest to the user. Letizia observes the user behaviour and builds the model based on these observations. Similarly user modeling system are Syskill and Webert [9] in which users are encouraged to rate Web pages and then a profile is generated of each user’s specific interests.

In relation to the application of user models the most investigated application area is related to Adaptive Hypermedia (AH). As mentioned above AH Systems use a user model to provide a more personal experience by adapting the content such as its presentation or navigation to the user’s needs and preferences [2]. One of the most successful application areas for AH mechanisms has been in e-Learning [3]. In the following three different adaptive e-Learning systems are introduced.

The Adaptive Hypermedia for All (AHA!) system, as a first example, is an open source adaptive hypermedia platform primarily used for educational purposes [10]. The motivation of AHA! is to adapt the learning content to learners with different background. The architecture of AHA! is web server based and combines domain/adaptation model and a user model. The domain/adaptation model contains the conceptual structure of the application along with the adaptation rules created by course authors; the user model represents the user's knowledge in the form of the number of times each conceptual model has been visited. Adaptation types supported by AHA! are adaptive link hiding and annotation, adaptive link destinations, and conditional inclusion of fragments [10].

Adaptive Personalized e-Learning Systems (APeLS) [11], as a second example, is driven by learning content model, narrative model, and learner model. Learning content model is a description of actual learning content in metadata format. Narrative model contains concepts of learning outcomes along with learning strategy. Narrative model and content model are mapped via metadata vocabulary. Learner model represents the knowledge of a learner based on the learner's interaction with the system. An adaptive engine processes the three models at run time to dynamically generate learning content for a learner. Extra models such as device model which allows adaptivity based on user's device can also be added if required. Similar to SEOW APeLS separate the different concerns such as content, user and pedagogical model in different components. Moreover APeLS applies a multi-model approach to allow personalization throughout diver's user groups.

KnowledgeTree, as a third example, implements an architecture for adaptive e-learning based on distributed reusable learning activities [12]. Similar to APeLS, the aim of the architecture is to fill the gap between the large amount of resources in the existing Learning Management System (LMS) and the potentially powerful adaptive systems to provide a new generation of e-learning systems featuring personalized content. The architecture consists of four core components: activity servers, value-adding services, learning portals, and student model servers [12].

However all three discussed systems are depending on a closed corpus scenario in which the content, the user model and the adaptation system is strongly interlinked. Approaches to drive Adaptive Hypermedia such as presented by SEOW have been introduced in several research projects.
SemWeb [13] as one example dynamically links and personalizes web pages by using linked data to annotate different elements in the viewed web page. However the personalized highlighting provided by SemWeb is generated on the client’s browser side and not server side as implemented with SEOW.

In relation to using web-based Content Management Systems (WCMS) to allow a more personalized experience several WCMS projects such as Semantic Drupal\(^5\) and the Semantic Web extension for MediaWiki \(^6\) have been introduced. These project provide extensions to the core WCMS implementation such as SEOW although focus more on the annotation of content and not on the personalized delivery. However based on developments related to the web of data as well as the introduction of RDFa\(^7\) to WCMS interesting approaches for a more personalized delivery of web based content can be expected allowing a more easy application of AH mechanisms on open data.

VI. CONCLUSION AND FUTURE WORK

The evaluation of both AMS and SEOW has indicated that by using a user model in connection with linked data to identify related concepts and terms and by using a simple WCMS extension, a more personalized and effective browsing experience can be provided.

The evaluation of AMS has indicated that it is an effective approach to implicitly model the user’s browsing interest. However AMS may show limitations in assessing interest across multiple domains by the lack of further information on the captured key terms. This limitation can be overcome by introducing term clustering and abstraction based on link data term hierarchy extractions.

SEOW on the other hand has been identified as a system which can use user model information to provide the user with a more personal browsing experience. The main design differences between AMS and SEOW are there location within the overall architecture with AMS being purely client sided and SEOW server sided. The limitations of SEOW are related to the need of domain related adaptive rules to identify terms which are relevant in the domain of interest. For SEOW to be used across multiple domains a large set of domain specific adaptive rules need to be designed. However based on assertions made from linked data analysis a semi automatic mechanism to author these adaptive rules could be provided in the near future.

Future work should also include the notion of cross site adaptivity allowing a personalized approach not only in one WCMS, but across several different instances of WCMSs. This notion has been introduced although no fully evaluated system has yet been discussed \([14]\).

ACKNOWLEDGMENT

This research is supported by the Science Foundation Ireland (grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (www.cngl.ie) at Trinity College, Dublin.

REFERENCES


