

Full Length Research Paper

Improving Web Content Management System: Template personalization approach

Hebah ElGibreen^{1*} and Samir El-Masri²

¹Information Technology Department, College of Computer and Information Sciences, King Saud University, Saudi Arabia.

²Information Systems Department, College of Computer and Information Sciences, King Saud University, Saudi Arabia.

Accepted 6 November, 2012

Web Content Management System (WCMS) has been developed to facilitate different aspects of website development. However, up until now, designing of web pages is accomplished by the website developers. Even though WCMS might suggest a number of templates, but it does not consider the nature of the website that will be developed. Many designers have complained about designing the right structure, and different users have complained about poor website structure. Thus, this manuscript will present a new approach that extracts personalized templates by applying web personalization in order to facilitate WCMS and increase its scalability and flexibility. In specific, a new technique that mixes hybrid Evolutionary Algorithm (Genetic Algorithm and Ant Colony Clustering) with Cluster Tree Matching is developed. Moreover, an experiment has been conducted to prove the quality of the proposed approach, which confirmed the good quality of the algorithm concerning speed, precision, and accuracy.

Key words: Web personalization, Web Content Management System, group profiling, evolutionary algorithm, document object model.

INTRODUCTION

Nowadays, the Web emerged and became the most-wanted method for information sharing and communication. Websites grow daily to serve different visitors for different purposes with its own structure. Due to such a rapid and hectic growth, Web access and effective design have become a challenge. It has been noticed that websites that have poor information and structure makes the visitors get lost and feel disoriented (Eirinaki et al., 2006). To respond to such challenges, web personalization has emerged to better fit the information access and design to the user's needs (Shen et al., 2006). Web personalization can be defined as providing the right content to the right person on behalf of the user.

Originally, web personalization was used in advertisement and promotion personalization for different

visitors. Currently, it focuses more on visitors to provide the appropriate information and service access to make the websites more useful (Flesca et al., 2004). However, website developers are usually neglected when it comes to personalization, such that, web personalization is used to personalize websites based on visitors' needs and none for the developers of such websites (Eirinaki et al., 2006). It mainly focuses on visitors' behaviors and concentrates on what contents should be presented to them during their navigation; while developers have to customize everything by themselves to build the website before it can be used by the visitors. Note that visitors are the users of the website while developers are the builder and owner of the website. For example, in www.Amazon.com, one type of the visitors is the customers who purchase books while the developers, on the other hand, are the people that built the page of purchase.

Web Content Management Systems (WCMS), on the other hand, is a tool that allows a variety of centralized

*Corresponding author. E-mail: hjibreen@ksu.edu.sa.

and decentralized non-technical people to manage, edit, create, and control a large and dynamic collection of HTML contents (He and Chen, 2010). Thus, it concentrates more on website developers' needs rather than the visitors of the website. However, current WCMS provides customization of web pages instead of personalization; such that developers customize the web page design by themselves. Sometimes it provides template samples, but such samples do not automatically consider the need of developers or the purpose of the website to be developed.

Accordingly, using web personalization in WCMS can help both WCMS and website developers. When using web personalization, WCMS developers would not have to provide a large repository of templates and, thus, no need for studying of templates' usability or updating it every period of time. Additionally, using personalization will reduce the website developers' time for checking similar websites and studying the visitors' needs to choose which structure is more appropriate for them. For example, if the website is for children, then WCMS can retrieve the template of other similar websites and, most probably, retrieve templates with simple structure that uses buttons instead of drop down menus. Moreover, no matter how much WCMS providers searched, studied, and analyzed web pages it is difficult to guarantee covering all cases, especially since the web users' demand is very dynamic and changes excessively. Thus, instead of studying Human Computer Interaction strategies, repeatedly, and searching for usable page structure, using personalization will automatically take advantage of others' experiences.

To further motivate the use of personalization in WCMS, it has been found that the spread of the World Wide Web and the increase of website development increased the necessity for WCMS to tailor itself for different website developers designing preferences. However, it has been noticed that most of the studies conducted to improve WCMS have only concentrated on increasing the usability without the flexibility or scalability. On the contrary, using personalization in WCMS will increase its scalability and flexibility; by automatically updating the templates based on the current trends and matched users' needs.

Regardless of the advantages stated above, there are some challenges when personalizing a web page template in WCMS. The most challenging factor is time. New users must be profiled and compared with all others and matched with similar ones to collect the suitable web pages. In addition, after matching, collected web pages must be further refined to extract and generate templates. Such process may take some time, especially if the number of users is huge. Moreover, the quality of the matching measure, which is used to match the users, is also important. Appropriate similarity measure must be chosen to ensure that the personalization process is accurate. Moreover, random modification, in color and

images, of the extracted templates must be considered to guarantee a degree of individuality.

Consequently, this manuscript will present a new approach to improve WCMS using web personalization; such that WCMS can take advantage of previous developers' knowledge to offer different template designs for each website developer depending on their needs. Hence, static content of websites (that is, templates) will be automatically built without the intervention of website developers or WCMS. Such an approach merges web personalization with automatic template extraction to improve the quality of WCMS and increase its flexibility and scalability. It mixes hybrid Evolutionary Algorithm (Genetic Algorithm and Ant Colony Clustering) with Cluster Tree Matching in order to accomplish the process of personalization.

Ultimately, it can be said that the proposed approach will increase the website's productivity and WCMS scalability and flexibility by considering the static content of web pages to personalize template extraction and, thus, automate the process of website design. Nevertheless, it must be noted that this manuscript is an extended version of a short paper that has been published as a work in progress by ElGibreen and El-Masri (2012). However, sufficient details of the application, method, analyses and results achieved is indicated and explained in this manuscript.

LITERATURE REVIEW

In this part of the paper, the main aspects needed to understand the proposed approach are defined. First, Web personalization is explained and its steps are discussed. Then, user profiling is presented. After that, HTML web pages are defined, and its representation is described. Finally, Evolutionary Algorithm is identified and some of its main techniques are shown.

Web personalization

In Eirinaki et al. (2006) web personalization has been defined as *"any action that adapts the information or services provided by a web site to an individual user, or a set of users."* Personalization, in general, is different from customization. In customization, the effort is passed on to the users, that is, they have to choose what is appropriate and what is not. However, in personalization, all the work is done automatically on behalf of the users, based on their data. It uses different techniques to collect users' data and then personalize the services based on the similarity between the current data and other content or users of the website.

There are a lot of companies that offer and use web personalization in their websites. For example, Yahoo.com personalizes their online advertisement

based on the users' behavior. If the user's log files showed that their interest is mostly in sport websites, then most of the advertisement ads will be focused on that area.

In general, web personalization requires a lot of efforts to gather the required information and data. It mainly includes four elements, as illustrated in Figure 1. First, web data must regularly be collected to be pre-processed and modeled. Then, such data must be analyzed and matched to finally determine what action should be performed (Eirinaki and Vazirgiannis, 2003).

In web personalization, different types of data are collected and used in the process of personalization. In specific, personalization data has been divided into four main categories, as follows (Eirinaki and Vazirgiannis, 2003). First, "Content" which is the directly retrieved data, such as text, images, or database information. Second, "Structure" which contains the content organization data, such as HTML tags and hyperlinks. Third, "Usage" which is the website usage data included in the Web access log, such as date and time of access, visitor IP address, and access paths. Finally, "User Profile" which is the website users' data, such as demographic information, preferences, or interests. Regardless of the type of data used, such data are collected in three different manners (Hildebrandt, 2008) depending on the type of the website and its visitors. It can be collected explicitly, that is, collected data are available and directly derived from its sources; or implicitly, that is, collected data are indirectly deduced from observing the user behavior; or in a hybrid manner where the collected data are gathered implicitly and explicitly.

In addition, the analysis of users' data to recommend a certain personalization has been categorized into four types: user-based, item-based, Web usage mining, and rule-based (Eirinaki and Vazirgiannis, 2003; Flesca et al., 2004). User-based (collaborative) personalization recommends and personalizes the website based on previous similar users' preferences. Item-based (content-based) personalization, on the other hand, recommends and personalizes the website for each user separately by relating their preferences with the available content. It recommends items similar to items previously liked by the user. Alternatively, Web usage mining analyzes and mines the Web log data to extract users' navigational behavior. Finally, rule-based personalization asks the users a series of questions, which are derived from a decision tree, to tailor the final result based on their answers. The focus of this manuscript will be on user-based personalization for the personalization process.

User profiling

Discovering users' difference is essential to provide the required personalized service. User profile has been defined in Hildebrandt (2008) as *"the process of 'discovering' correlations between data in databases that*

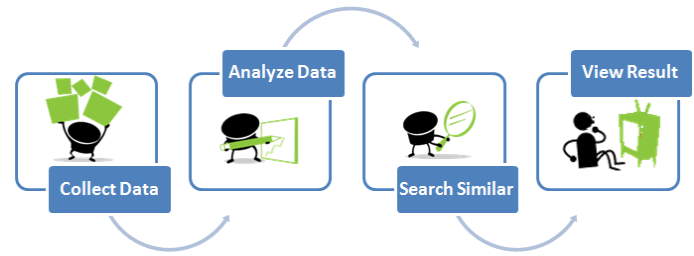


Figure 1. Web personalization elements.

can be used to identify and represent a human or nonhuman subject (individual or group) and/or the application of profiles (sets of correlated data) to individuate and represent a subject or to identify a subject as a member of a group or category." In general, user profiles vary in content and type, which usually depend on the application. The content gathered can be user interests, knowledge, background, skills, goals, behaviors, interaction preferences, characteristics, or context. However, the profile type itself can be grouped or individual, public or private, static or dynamic, distributed or non-distributed, explicit or implicit, feedback or stereotypes. These content and profile types have been discussed and further explained in Crossley et al. (2003) and Hildebrandt (2008). In addition, in order to build and use profiles, different techniques have been developed and further improved in the last few years (Schiaffino and Amandi, 2009), such as machine learning, genetic algorithm, and classification techniques.

Hypertext markup language (HTML) web pages

In order to understand how web page templates can be automatically extracted it is important to know the main aspects of HTML web pages and how it is handled. In general, web pages include static and dynamic content (Ravi et al., 2009). Dynamic content usually changes depending on the visitors' activities and does not have a fixed design or definition. On the other hand, static contents are the content that are usually fixed during the visitor navigation, and are usually designed by the website developers with the help of WCMS. Hypertext mark-up language (HTML) is one language that is used to define the static content. It has been defined in Wan Mohd Mahidin (2003) as *"a web programming language used to display web pages."* Usually, browsers provide web pages based on HTML files, which define the structure and represent the pages as tags and attributes. Thus, different web pages can be generated by extracting some nodes of the HTML tree (Christos et al., 2004).

In order to navigate, modify, add, or delete content or elements of HTML web pages, Document Object Model (DOM) has been developed by W3C as HTML standard

interface ("What is the Document Object Model?," 2004). Specifically, HTML files are represented by DOM trees in order to apply different similarity measures and extract common templates (Chulyun and Kyuseok, 2011). DOM has been defined as "a standard for accessing documents, like HTML" and "presents an HTML document as a tree structure" (Chulyun and Kyuseok, 2011). Thus, DOM can be used, with any programming language as an interface to HTML web pages. Its representation aims to display and show the structure of HTML web pages rather than its content.

For example, Figure 2 illustrates how HTML tags can be represented in a DOM tree. The hierarchy of the tags is preserved, and the content of the page is stored at the leaf nodes.

Templates are usually used to take advantages of the static content of previous web pages and define a general structure that standardizes such content (Laahs et al., 2008). HTML template extraction, in specific, has been divided into two categories: site-level and page-level (Chulyun and Kyuseok, 2011). Site-level template extraction extracts the template from different pages in the same site while page-level extraction, on the other hand, extracts the template from a single page. In this manuscript, the focus will be on page-level extraction for HTML template extraction.

Nevertheless, HTML web page structure must be compared and, thus, DOM tree similarity must be measured. The most famous tree similarity measure used in DOM trees are Tree Edit Distance (TED) (Bille, 2005) and Simple Tree Measure (STM) (Ferrara and Baumgartner, 2011). TED measures between two trees returns editing operations' cost of turning one tree to the other (Bille, 2005). Such measure introduces high complexity, especially with large trees. On the other hand, STM measures the similarity between two trees by mapping and comparing the nodes and return its maximum value (Kim et al., 2007). A node mapping value is equal to one, assuming equal weights of all the nodes. Using STM will simplify the process of matching since, in contrast of TED, no inserting, deleting, or re-labeling operations are needed. However, the assumption of equal weight is not valid in HTML trees because HTML tags have different values depending on the data browsed in the web page. Thus, STM measure has been improved, and a new measure called Cluster Tree Matching (CTM) (Ferrara and Baumgartner, 2011) has been developed.

CTM measure was developed based on STM by Ferrara and Baumgartner (2011) to calculate the tree similarity recursively in a matrix and count the weight difference between two trees. Their measure divides the greater number of siblings between two trees over the match value between two nodes, level by level; thus, it cluster the matching process sub-tree by sub-tree. It assigns a weighted value to give less weight to insignificant sub-trees; such that deep levels of HTML

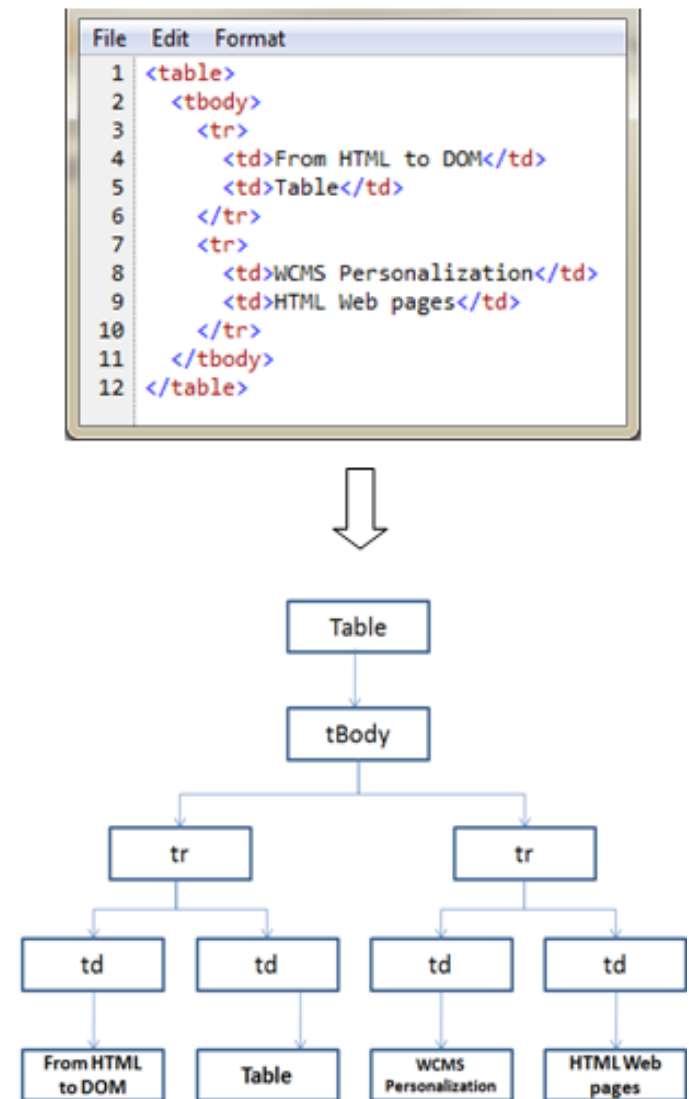


Figure 2. HTML to DOM tree representation.

trees usually indicate an insignificant component of the web pages like table rows or list of items. CTM increases the accuracy whenever the trees complexity and similarity increases. Thus, it is more effective than STM and less complex than TED. Figure 3 illustrates two trees (a) and (b) and circles the similar elements between them. The result of CTM will be equal to 0.375 (that is, 38%) which represent the division of 3 over 8. In this manuscript, the focus will be on CTM for tree matching, thus, more details will be explained later.

Evolutionary algorithms

Evolutionary Algorithms (EA) (Freitas, 2002) are artificial intelligent algorithms that have been inspired by nature. Such algorithms are used to solve any kind of problems

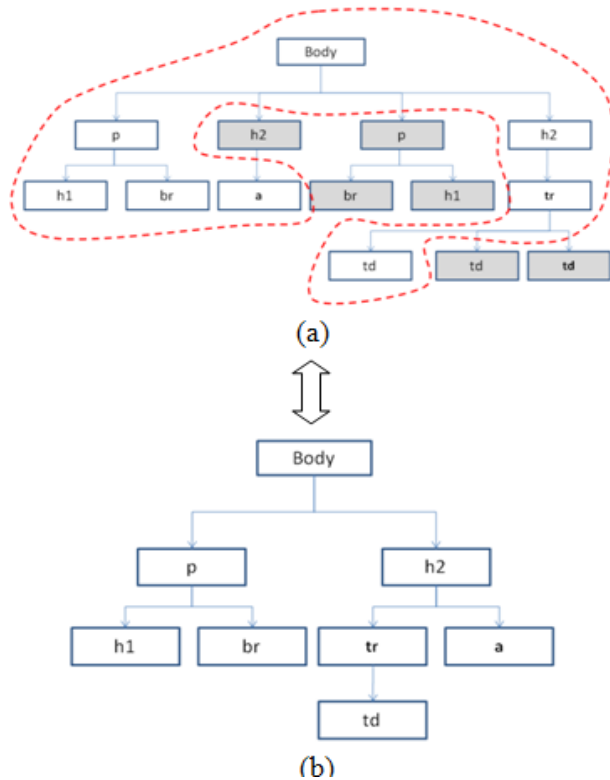


Figure 3. CTM tree measure.

and identify candidate solutions as fast as possible. It is useful when the best solution is unknown, and the data repository is very large. Different techniques have been developed under the umbrella of EA, such as Genetic Algorithm (GA) (Wook and Woo, 2005) and Ant Colony Clustering (ACC) (Deneubourg et al., 1991). Each technique has been tailored in different domains depending on the problem and solution required. In this manuscript, the focus will be on GA and ACC techniques for problem solving.

GA is considered as the most popular technique of EA. It is “a general purpose search algorithm which use principles inspired by natural genetic populations to evolve solutions to problems” (Wook and Woo, 2005). It typically starts from a collection of random solutions, called population, and then evolves it using selection and reproduction. The fittest solution is chosen based on the fitness function that measures the strength of the solution. The population contains elements, called individuals or chromosomes, to represent possible solutions. In GA, four steps are performed to match individuals (Russell and Norvig, 2003). First, it represents the available datasets to the required data type, such as text, character, or binary representation. Then it initializes the population by randomly assigning different values to the chromosomes. Afterward, it reproduces a new solution by mixing every chromosome with another to reproduce an offspring. Finally, it selects the best chromosomes based on their fitness value. Figure 4

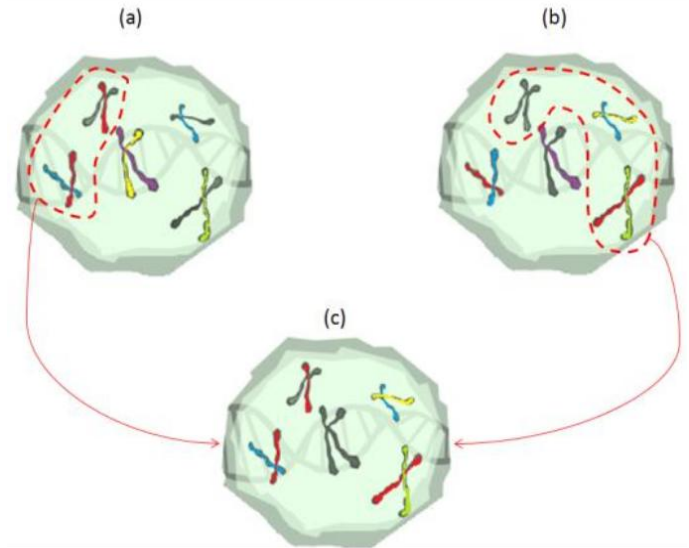


Figure 4. GA: (a) initialization (b) reproduction (c) selection.

illustrates how population of chromosomes starts reproduction, after the representation step, to produce a new solution and better results. GA can search potentially large spaces and, thus, it can solve the problem of large users' repository and improve the performance of profile matching, considering the time.

Alternatively, ACC method is a new type of clustering that is based on an algorithm called Ant Colony Optimization. Such an algorithm is an Evolutionary Algorithm that is built based on nature. In specific, it is developed based on how ants look for food by spreading their pheromone to find the food path (Wei et al., 2005). Figure 5 illustrates how the ants search for food. It starts updating its local pheromone (Figure 5b) and explores all possible roads to finally emphasize the shortest one. As a result, the global pheromone (Figure 5c) will be emphasized, while the other pheromone is fading, by communicating the local pheromone, that is, the best solution will be chosen.

ACC is developed to optimize the process of clustering and improve its time. It is mostly used in web usage mining (Lu et al., 2007) and basically divided into three steps (Dinh and Mamun, 2004). First, it updates the local pheromone, that is, update each ant solution with the suitable cluster. Then, it applies the state transition rule, that is, decide if a certain ant should be dropped in a cluster or out of it. Finally, it updates the global pheromone, that is, update the final solution which is the road map.

Related works in personalization and template extraction

To the best of our knowledge, no one yet has considered personalization in template extraction; it was treated

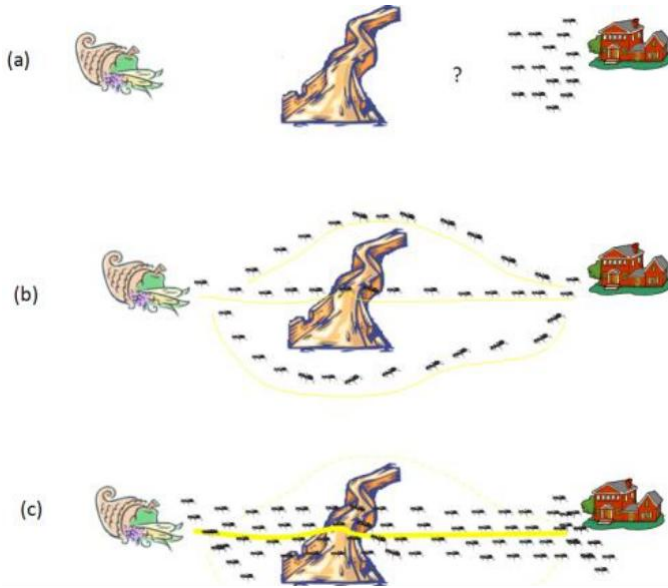


Figure 5. Ant colony optimization.

separately to handle dynamic content and website navigation. Thus, available work in personalization and template extraction is discussed here separately.

When it comes to web personalization, Yen and Kong (2002) proposed a Personalized Electronic Catalogue System to personalize e-commerce websites. Display, organization, and information content were personalized based on the interest and preference of each website visitor. Moreover, Flesca et al. (2004) proposed a new personalization strategy to use content and user interest similarity. The purpose of their strategy is to recommend web page navigation that is appropriate for each user. Thus, they presented a new methodology that allows users to personalize their web page navigation through websites by emerging content and usage mining techniques within the field of recommendation systems and analyzing users' preferences and related content in order to suggest the most relevant pages to users. Similarly, in Shen et al. (2006) web mining model that dynamically creates personalized web pages is presented. Their model was based on hidden Markov model and clustering; such that users with similar interest are grouped and their visited pages are related.

On the other hand, in Nasraoui et al. (2008) a complete framework is presented along with web usage mining of real websites. Their framework first profile users explicitly, using search query that is stored in the web log data, then the web content is described in the ontology, and the result is validated in terms of its adaptability to user behavior. Their profiling is done by clustering user sessions and group users based on their activities to extract profile of each group as a set of significant URLs. After the clustering process, website hierarchy is exploited to give similarity weight between URLs.

Moreover, for further improvement, a web personalization was proposed in Eirinaki et al. (2006) and applied to semantically annotate websites' content. They integrated content semantics with the user's activity patterns (navigational actions) and then represented both using ontology. Furthermore, to personalize the website, they used techniques of web mining and enhance the users' behaviors with semantic knowledge.

Additionally, in Munk et al. (2010), it was found that web log mining effect personalization because it is used to mine the websites visitors' activities and personalize the web based on in. Hence, techniques used for web log mining were evaluated to conclude a matrix that can be used to minimize the time of personalization. Moreover, Perugini (2010) tried to formalize the transformation of hierarchal website and built a toolkit to collect all possible transformations. The toolkit was used to make shortcuts to the desired information and to personalize the user navigation based on his previous interaction with the websites. In Rana (2012), however, a study was carried out to show the different web usage mining that is used to study the behavior of the visitors while interacting with a website. In Eslami et al. (2011), nevertheless, web services were personalized to allow non-technical users to use and create services in homecare domains without the need to go into the technical details of such services. Moreover, Carminati et al. (2012) proposed a multi-layer framework to personalize collaborative tagging, which is used as metadata when sharing online resources.

Alternatively, a new attempt of personalizing WCMS was found in CoreMedia (2012). CoreMedia Corporation has recently launched a new solution that was added to their WCMS. This solution allowed media and entertainment companies to personalize their multimedia content based on social, historical, situational, and behavioral of audience data. Such personalization would reduce the need for developers to constantly update their websites. However, again, this personalization is based on visitors' behavior and considers the dynamic content of a website, not the static.

Consequently, it can be noticed that all the previous studies of web and WCMS personalization focus on the behaviour of the visitors and how to manage the dynamic content of the websites. However, the needs of the developer who come up with such websites were neglected, and the static contents of web pages were usually customized.

Alternatively, when it comes to template extraction, it was found that template matching has been applied to different domains. For example, in Chávez et al. (2012), template were extracted from laser pointer and matched to other templates, stored in the system, in order to discover how to interact with smart home users and activate the demanded service using laser pointer commands. However, since the scope of this manuscript is about web template, the focus will be on template extraction in the web domain.

In Ji et al. (2010) a new method suggested parsing

different websites into HTML Tag trees to extract the content of pages by identifying repeated patterns. The Tag tree was constructed based on certain rules in order to parse HTML web pages into a tag tree then extract the template by allocating the content in the leaf nodes. On the other hand, in Gui-Sheng (2010) another method is presented to induce web page template. Their method suggested to first select websites randomly, and then transforms it to DOM tree, judge its similarity, and cluster the web page to extract templates, correlate it, and extract other pages' content. Moreover, in Haikun (2010) an approach of three steps was proposed. First, pages are transformed into DOM trees to be fed to a tree clustering module that calculates trees' similarity. Then, the resulting trees will be presented to users as pages so that users can select interesting data in order to finally generate the desired template for data extraction. In addition, it was introduced in Shui-Lung (2004) a Tree Template Automatic Generator to learn template from given web pages. The template is generated using a top-down approach starting from the root and going down one level at a time.

Regardless of all the evaluation conducted in the previous works, some important deficiencies have been noticed. When it comes to personalization, all the works discussed have only considered the visitor behavior, and the focus was on the website/visitor relationship; such that the problem of navigation is only considered, thus, website designer has been neglected in web personalization. In addition, when it comes to filtering and recommending process of web personalization, most of the available works used only content based similarity. Consequently, even though it is called web personalization but the focus was mostly directed to navigation personalization and content similarity.

Alternatively, when it comes to template extraction, the available works have directed the focus to the dynamic content of the website rather than the template itself. Templates were generated only to identify the content of different websites without taking advantage of the semi-structured¹ knowledge that has already been extracted. In addition, time of template extraction and its accuracy was not fully considered in these works.

As a result, this work next proposes a new approach which considers personalization for template extraction in order to simplify the process of website designing on the developers and to further improve WCMS. Hence, users' similarity, time of template extraction, and its accuracy were considered when developing such an approach.

WCMS TEMPLATE PERSONALIZATION APPROACH

WCMS template personalization is a new approach to personalize the process of template extraction in WCMS.

Such an approach captures both structure and user profile data by mixing hybrid Evolutionary Algorithm (EA), specifically GA and ACC, with CTM. First, it regularly collects data from the developers' account. Then, it creates individual profiles for matching. After that, when developers choose to design their websites, profiles will be matched and common templates will be extracted. The steps of the proposed approach are illustrated in Figure 6 and its details are explained next.

Data collection

The data used to build the individual profiles are collected explicitly from the developer account. Such data would contain the developers'² country, religion, and their business scale (large company, small company, individual). In addition, more data are collected about the type of visitors, in which the website is targeting, including age (kid, adult, elderly), language, gender, ability (normal, mute, autism...), and nature (business, discussion forum, medical...).

Individual profiling

After collecting the required data, profiles will be generated. The profile of each website will be gathered in Extensible Markup Language (XML) file. XML is a standard language for describing the data in a tree structure. Thus, using such structure will increase the scalability of the proposed approach for any field, and it will be possible to profile any website as long as it has such XML tags in its metadata. The profile will contain the characteristics collected in the first step; such that each tag is a characteristic, and the data are its value. All websites that target the same age are grouped together in order to simplify and speed up the process of matching. An example of such a profile is illustrated in Figure 7.

It must be noted that the collected data can change in time. However, since these changes are usually not frequent then it is considered as a long-term interest. Thus, every period of time the first and second steps of the proposed approach are repeated, in the background, in order to consider the possible changes in the long-term interest without affecting the personalization speed.

Profile matching

Whenever the developers access WCMS, in order to design the website interface, their profile is matched with all other users to personalize templates. Genetic

¹ Semi-structured documents are documents that have a known content without knowing where such content is placed in the document.

² When the word developer (or user) is said alone in this paper it would mean website developer.

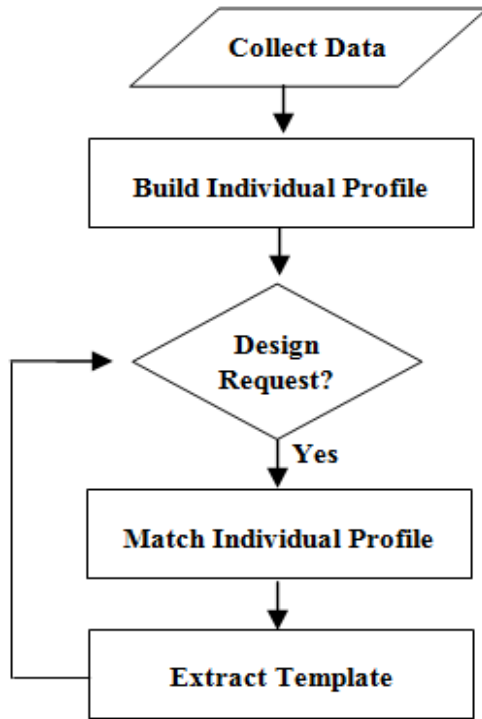


Figure 6. WCMS template personalization approach.

```

<?xml version="1.0" encoding="ISO-8859-1"?>
- <UserProfile age="kid">
  - <User ID="123456">
    <country>Any</country>
    <religion>Any</religion>
    <scale>Individual</scale>
    <language>Arabic</language>
    <gender>Any</gender>
    <ability>autism</ability>
    <nature>medical</nature>
  </User>
  + <User ID="123459">
  + <User ID="123460">
</UserProfile>
  
```

Figure 7. Individual profile.

Algorithm is used to accomplish this step. It typically starts with random solutions then evolves it by repeated selection and variations of more fitted solutions. The elements of the population (chromosomes) are used to represent candidate solutions; for example, if the chromosome size is 4, then it can be represented as illustrated in Figure 8. The content of this chromosome

"123456"	"123789"	"654321"	"987654"
----------	----------	----------	----------

Figure 8. Solution chromosome.

represents the similar users' ID, that is, possible solution, and each gene in the chromosome will be linked to the related characteristic extracted from the XML file generated previously.

In order to produce the most fitted solution the pseudo code, illustrated in Figure 9, is applied. First, the population will be initialized by users of the same age, given the maximum number of genes; such that each chromosome will represent a solution. The maximum number of genes represents the maximum number of similar users who can be considered. Such number is specified because the repository of profiles can be very large, which tremendously increases the chromosome length to the extent that might affect the process time without much of improvement in the solution.

The fitness value of chromosome I is calculated using Equation (1), where $D(I, T)$ is the similarity distance between the targeted user profile T and other users included in the chromosome, G is number of genes, and Ch is number of characteristic in profiles. The division used in calculating the fitness is introduced to scale its value between $(0...1)$ and, thus, simplify the matching process. The similarity measure $D(I, T)$ is calculated using Equation (2) for all genes in the chromosome. $ChMatch$ is a function that calculates the similarity between two profile characteristics using Equation (3). As a result, when the similarity distance between the users increases then fitness value will also increase and, thus, the resulting solution will be more fitted.

$$F(I) = \frac{D(I, T)}{[G \times Ch]} \quad (1)$$

$$D(I, T) = \sum_{i=1}^G \sum_{w=1}^{Ch} ChMatch(I_{iw}, T_w) \quad (2)$$

$$ChMatch(I_w, T_w) = \begin{cases} 1 & \text{if } I_w = T_w \\ 0 & \text{if } I_w \neq T_w \end{cases} \quad (3)$$

After calculating the fitness of all users in the chromosome the population is reproduced and tested until the stop condition is reached or target solution, where the fitness is equal to one, is found.

Template extraction

In this step, templates are extracted depending on the


```

Match Profile {
Input: users profile U, target user T, population size:  $\beta$ , stop condition  $\alpha$ , chromosome size  $\gamma$ .
Output: Solution S.

//Initialize population and calculate the fitness of each solution
P = Generate Population (U, T,  $\beta$ ,  $\gamma$ );
S = best solution in P;

While ( $\alpha$  not reached) and ( $F(S) \neq 1$ ) do
    //Crossover to produce better solutions
    Pnew = Reproduce (P)
    Calculate Pnew fitness;
    //Selection
    P = Select best  $\beta$  chromosomes from P and Pnew
    S = best solution in P;
End while
Return S; }

```

Figure 9. Profile matching.

solution generated previously. Note that profiles matching and template extraction steps are repeated every time the users want to redesign their website because users and visitors' interest might change, that is, changing in the short-term interest is considered. This extraction is further divided into three steps, as follows.

Hypertext markup language (HTML) parsing

Parsing web pages into trees will simplify and speed up searching in such pages. Hence, in order to simplify the process of web pages matching, each website front page is parsed into a tree using DOM representation. In the proposed approach HTML documents are parsed to DOM trees where each node in the tree represents an HTML tag and associated with its attribute, if exist. Since only the template is extracted without the data then attributes and leaf nodes, which contain the text, will be removed to reduce the space. In addition, each tree will be identified by the developer (user) id and his profile fitness value to be used later. An example of the resulting tree is illustrated in Figure 10.

Tree clustering

After parsing HTML pages into DOM trees, the resulting trees are clustered into groups to identify similar templates and avoid redundant results. To accomplish this step ACC is used with CTM. In this step, each tree represents an ant, and the clusters represent the roads. The solution is represented by a vector with N elements and W values; such that N represents the number of trees, and W represents the number of clusters. For

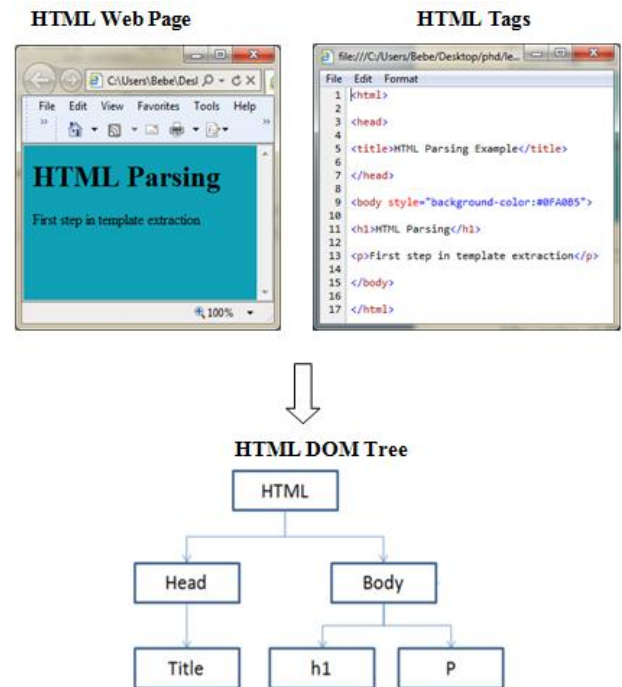


Figure 10. Tree generation.

example, if the number of trees extracted were five and the number of clusters given are three, then a solution could be as illustrated in Figure 11, which means that the second, fourth, and fifth trees belong to the same cluster while the first belong to cluster# 2, and the third tree belong to cluster# 3.

After initializing all the attributes, this step works as follows. First, each tree will be assigned randomly to a

T1	T2	T3	T4	T5
2	1	3	1	1

Figure 11. Cluster solution.

cluster, that is, ants are positioned randomly on the road. Next, while the stop condition is not reached, each tree is matched with all other trees, that is, each ant searches its neighborhood to find the most similar one. The similarity measure between two trees T_i and T_j is calculated using the TreeMatch function illustrated in Figure 12; where $d(n)$ is the node n degree in a tree (number of first-level children), $T[i]$ is a sub-tree of tree T starting from level i and afterwards, $s(n)$ is the total number of siblings of a node n including itself, and M is the measure matrix. Such measure selects sub-trees, which share the same root, and analyze its tag name to return the similarity weighted measure between two trees. TreeMatch function is based on CTM, discussed previously, which calculates tree similarity recursively in a matrix and counts the weight difference between two trees.

After calculating the similarity measure it will be possible to apply the transition rule in order to update the local pheromone and decide whether to pick up a neighbor cluster or drop the current tree cluster. Usually, in the original ACC algorithm, if the current ant does not carry an object (cluster) while its neighborhood has an object, then it will take the same object as them based on a pickup probability, that is, join their cluster. On the other hand, if the current ant found to be carrying an object but its neighborhood carries a different object, then it will drop its object based on a drop-down probability, that is, it will be temporary with no cluster.

However, it has been found that such manner, in WCMS personalization, could affect the accuracy, especially if the global solution was initialized with no cluster. Thus, a slightly different transition rule has been exploited and used to improve the accuracy, as follows. If the current tree has similar clustered neighborhood then the neighborhood cluster will be picked up, based on probability P_p , and the current tree will join the same cluster. On the other hand, if the current tree has a similar cluster to none neighborhood tree, this means that the clustering is inaccurate; therefore, the target tree must drop down its cluster, that is, will not belong to a cluster, based on a certain probability P_d .

The probability of picking up (P_p) and dropping down (P_d) a tree out/in a certain cluster are calculated using Equation (4) and (5), respectively, where f is the similarity measure calculated previously, using TreeMatch function, K_p and K_d are threshold constant defined at the beginning. If f is much higher than K_p then the measured tree is already surrounded by similar neighbors, that is, the probability will be close to zero and such tree will not

TreeMatch {

Input: two trees $T1$ and $T2$.

Output: weighted similarity measure W .

//if root nodes then it should not matches

If $T1.root \neq T2.root$

Return 0;

//if not root then calculate the weighted similarity

Else

$m = d(T1.root);$

$n = d(T2.root);$

// Initialize measure matrix

$M[i][j] = 0$ for all $i = (0 \dots m)$ and $(j = 0 \dots n);$

//start matching sub-trees

For $i = 1$ to m do

For $j = 1$ to n do

// Recursive call to measure sub trees similarities

// starting from current node as root

$SupTreeM = TreeMatch(T1[i-1], T2[j-1]);$

$M[i][j] = \text{Max} (M[i][j-1], M[i-1][j], (M[i-1][j-1] + SupTreeM));$

End for

End for

//Check if leaf nodes is reached

If $m \neq 0$ and $n \neq 0$ then

$W = \frac{M[m][n]}{\text{Max} (s(T1.root), s(T2.root))};$

Else

$W = \frac{M[m][n]+1}{\text{Max} (s(T1.root), s(T2.root))};$

Return W ;

End if }

Figure 12. Tree matching.

pick up a new cluster. Alternatively, if f is much higher than K_d then the result will be close to one and, thus, the probability of dropping down the tree cluster is low because it must exceed P_d . In addition, if the similarity between two trees is zero while they have the same cluster then P_d will be equal to 1 in order to force the drop down of the cluster, since there are no shared tags between the two trees, that is, they should never belong to the same cluster. At the end, after comparing all trees together and updating their local pheromone, the global pheromone will be updated. Then, the process will be

repeated in order to further improve the solution and assign trees to the most appropriate cluster or group.

$$P_p = \left(\frac{K_p}{K_p + f_{(i,j)}} \right)^2 \quad (4)$$

$$P_d = \begin{cases} \left(\frac{f_{(i,j)}}{K_d + f_{(i,j)}} \right)^2 & f \neq 0 \\ 1 & f = 0 \end{cases} \quad (5)$$

It must be noted that only one tree can update the global pheromone in each round to guarantee testing as many roads as possible. In addition, some trees might not find any similar neighborhood; thus, it has been decided to leave such trees in a separate cluster to ensure that its characteristic does not disappear. Doing such separation will ensure that unique web pages are preserved to improve the clustering in case a similar website has joined the Web. The pseudo code of the proposed clustering algorithm is illustrated in Figure 13.

Template generation

At the end, in order to extract the common templates for the developer, the resulting clusters will be examined and trees with the highest fitness in each cluster will be chosen. After that, attributes such as images and colors will be changed to random values. Then, DOM trees are parsed back to HTML files to be viewed as the personalized template.

IMPLEMENTATION

In order to test the performance of the proposed approach an experiment has been conducted. Here details of the experiment, explanation of its result, and highlights of the main issues are discussed.

Experiment settings

The proposed approach has been implemented with Java language in JBuilder environment. It was executed on a PC with Intel®Core™ i7 CPU, 2.67 GHz processes, and 6 GB RAM. In addition, different websites with different languages and characteristics have been used in the experiment. Each website has a full profile which was synthetically built in a database. The websites used to build such database are listed in Table 1. However, it has been found that almost all the websites have an unclear structure which causes some errors during the execution. Thus, an external library, called HTML Cleaner (net.sourceforge.htmlcleaner, 2010), was used to extract

```

TreeMatch {
Input: two trees T1 and T2.
Output: weighted similarity measure W.
//if root nodes then it should not matches
If T1.root ≠ T2.root
    Return 0;
//if not root then calculate the weighted similarity
Else
    m = d(T1.root);
    n = d(T2.root);
    // Initialize measure matrix
    M [i][j] = 0 for all i = (0 .. m) and (j = 0 .. n);
    //start matching sub-trees
    For i = 1 to m do
        For j = 1 to n do
            // Recursive call to measure sub trees similarities
            // starting from current node as root
            SupTreeM = TreeMatch (T1[i-1], T2[j-1]);
            M [i][j] = Max ( M[i][j-1], M[i-1][j], (M[i-1][j-1] + SupTreeM) );
        End for
    End for
    //Check if leaf nodes is reached
    If m ≠ 0 and n ≠ 0 then
        
$$W = \frac{M[m][n]}{\text{Max}(s(T1.root), s(T2.root))};$$

    Else
        
$$W = \frac{M[m][n]+1}{\text{Max}(s(T1.root), s(T2.root))};$$

    Return W;
End if }

```

Figure 13. Tree clustering.

the websites from its URL and clean the web pages, and then converts it to HTML pages.

During the experiment, all the parameters were unchanged, except for the stop condition, including population and chromosome size, pick up and drop down probability, in addition to the number of clusters. Population and chromosome size was set to "10" because only ten templates are wished to be viewed. In addition, the pickup probability (K_p) and drop down probability (K_d) have been initialized with "0.5" because it's unknown if it is better to drop or pick the cluster, thus, 50% chance is given in both cases. Lastly, the maximum number of group was set to "5".

Here, ten experiments, that have different stop conditions starting from 10 to 100, have been recorded to measure the performance of the proposed approach, where the templates are extracted for news company website that is written in English and target normal users. In each experiment, different measures have been calculated and recorded, as identified and next explained.

Evaluation measure

In order to evaluate the result of the experiment three types of measures have been calculated. The first

Table 1. Website samples.

Website
http://drupal.org
http://epda3.net/iv/mnshd-40.html
http://fcsn.org/index.php
http://games.sh3bwah.maktoob.com/3-0
http://googleblog.blogspot.com
http://gulfkids.com/ar/
http://www.children.gov.on.ca/htdocs/English/topics/specialneeds/index.aspx
http://kids.nationalgeographic.com/kids/
http://kids.yahoo.com/
http://news.google.com
http://news.maktoob.com/
http://pbskids.org/
http://science.nasa.gov/kids/
http://shopping.kelkoo.co.uk
http://wordpress.org
http://www.6abib.com/
http://www.abs-cbnnews.com/
http://www.adabatfal.com/
http://www.adpsn1.netfirms.com/
http://www.aljazeera.net/portal
http://www.Amazon.com
http://www.arabnews.com/
http://www.Ashford.com
http://www.atfaal.net/
http://www.bbc.co.uk/news/
http://www.delicious.com
http://www.ebay.com
http://www.facebook.com
http://www.feedo.net/disability/DisabilityMain.htm
http://www.Finance.yahoo.com
http://www.google.com
http://www.healthofnations.com/?gclid=CJ2J0_3m0qgCFclMfAodL0qNgQ
http://www.housesforrent.ws
http://www.joomla.org
http://www.kids.jo/main/
http://www.kidscom.com/
http://www.msnbc.msn.com/
http://www.news.yahoo.com
http://www.officedepot.com
http://www.phpbb.com
http://www.q8sneed.com/
http://www.rentals.com
http://www.s9y.org
http://www.shopping.yahoo.com
http://www.simplerachines.org
http://www.time4learning.com/index.htm
http://www.werathah.com/special/index.htm
http://www.zdnet.com/news
http://www.bbc.com
http://health.yahoo.net/

measure is the algorithm speed, which record the time of executing the profile matching and template extraction phases of the algorithm, that is, the personalization step. The second measure, however, records the quality of the individual matching using the fitness of the extracted solution in each experiment to indicate how much the users included in the solution are similar to the target user. At the end, the clustering algorithm quality was assessed using two measures: inter-cluster and intra-cluster similarity.

Inter-cluster similarity measures the similarity degree between pages in the same cluster using TreeMatch function illustrated in Figure 12 and then divides it by the number of users in each cluster to finally divide the total of all clusters over the number of cluster; such divisions are used to scale the final value from 0 to 1. Of course if the result of such a measure is high, that is, pages in one cluster are very similar, this will indicate good quality. On the other hand, intra-cluster similarity measures the similarity between the clusters. The average pair-wise distance has been used to calculate the similarity between the clusters; which measures the distance between the most fitted trees of each cluster all together using TreeMatch function, illustrated in Figure 12, and then divide it by the number of cluster to scale its value from 0 to 1. Naturally, if the result of this measure has a low value, that is, each cluster is very dissimilar from the other; this will indicate good clustering quality. The result of these measures and its details are further discussed.

DISCUSSION

After conducting the ten experiments, using the parameters specified previously, each measure is calculated and recorded. In Figure 14, the execution time of the approach is illustrated. As it can be noted, when the stop condition increases the time will most probably increase. Such result is normal because the profile matching and tree clustering algorithms depend, partially, on this condition. Nevertheless, the time taken to execute the algorithm, in any experiment, is very low. Most of the experiments take less than one second while the maximum time it reaches was less than second and a half. This indicates that the algorithm performance, in terms of speed, is very good.

The good result was achieved because the proposed algorithm refines and normalizes the data before going into the personalization. Specifically, the dataset is refined based on the profile of the webpage that is stored as XML tags. Hence, searching in the tree data structure of XML and matching profile would not consume a lot of time. Moreover, reducing of the dataset based on the stored profile would tremendously reduce the personalization process and, consequently, increase the algorithm speed. Hence, even if the dataset was large, the proposed algorithm will reduce such data and work on the related web pages only. Hence, the proposed

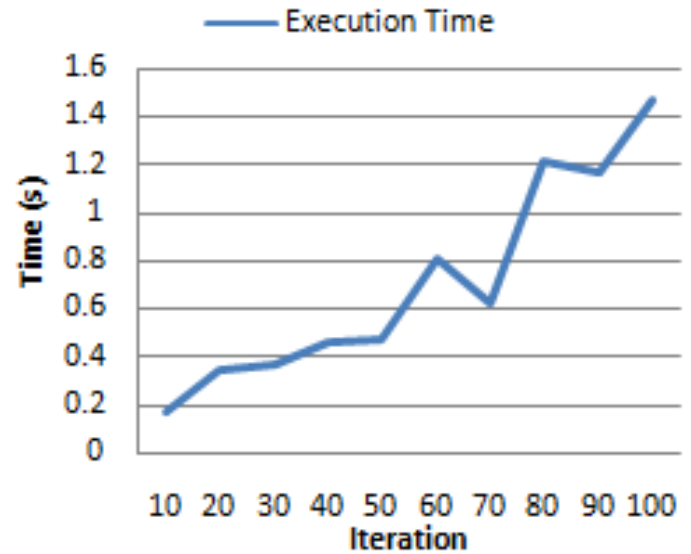


Figure 14. Algorithm speed.

algorithm can be scalable over a large dataset.

Additionally, the personalization precision of the proposed algorithm was measured using the quality of the GA resulting solution, as illustrated in Figure 15. Hence, the performance of profile matching step is recorded in each experiment. As it can be noted, the increase of the stop condition did not make any major differences in the fitness value. In fact, it started to slightly decrease the quality of the solution. Nevertheless, the quality of the solution extracted in each experiment was, on average, 98% match with the target user. Thus, it can be stated that the quality of the algorithm, concerning the profile matching, is high but a fair stop condition value is recommended.

Finally, in order to measure the personalization accuracy of the proposed algorithm, clustering quality were recorded. In specific, inter/intra cluster similarities have been measured and recorded in each experiment, as illustrated in Figure 16. Inter-cluster similarity indicated high percentage (99%) while intra-cluster similarity, on the other hand, indicated low percentage (0.1%). Such result denoted that the clustering algorithm is good; it revealed that such algorithm did not group dissimilar trees together and, also, did not separate similar ones. The performance excellence was because of the data refinement and profile matching, discussed before, where only the most related web pages will be considered in the clustering step. However, it must be emphasized on the fact that increasing of the stop condition did not affect the value of the clustering quality. Thus, a fair stop condition (equal to the number of pages) is also recommended.

After understanding and testing the proposed approach, it is time to summarize its contribution and compare it with others work. First of all, on the contrary of the available work discussed previously, the proposed

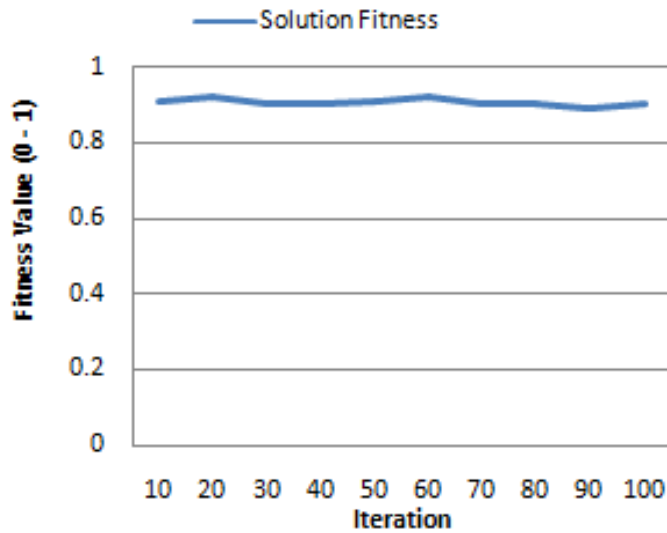


Figure 15. Personalization precision.

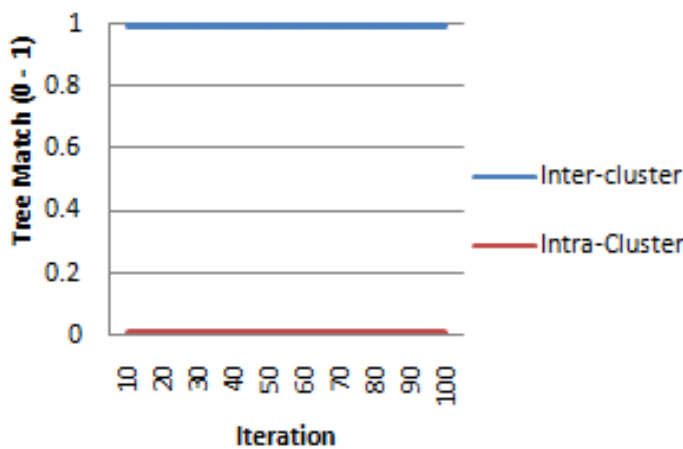


Figure 16. Personalization accuracy.

approach has further generalized web personalization. Instead of using such a personalization only on navigation and dynamic data it was applied on the static part of web pages to also consider the developer of websites rather than only the visitors. Thus, such an approach has added a value to the area of web personalization to further include providers' need, personalizing static content, and also consider the characteristics when matching rather than keywords.

Moreover, such an approach considered the long and short-term interest; it does not always update and re-execute the whole process. Only the matching and template extraction steps are repeated with every query; however, the rest is done once and updated every period of time while its result is stored in XML files. Such decision reduced the time tremendously without affecting the accuracy and, also, improved the scalability of

WCMS. In addition, the use of XML files has simplified and speeded up the process of profile matching. This is because XML content is represented as trees; thus, websites' profile will be easily searched, and information will be quickly fetched.

In addition, in order to handle web pages content, HTML was parsed to DOM trees. The uses of DOM trees have simplified and further improved the speed of web page matching and clustering. This improvement is because searching and matching trees is much better than going through the whole web page content sequentially. Moreover, such an approach is very dynamic due to the use of different parameters. Such parameters can be changed depending on the providers' need; for example, one provider could prefer only one template to be viewed; thus, the chromosome length in the matching step can be set to one. Additionally, pages were not forced to join a cluster to ensure that unique web pages are preserved in case similar websites join the Web.

Finally, as future work, ontology, which is a repository of knowledge, could be used when acquiring data; such that, instead of giving the developers limited options when entering their characteristics they can enter what they wish, and then it can be translated using the ontology's repository. For example, instead of giving the options (business, discussion forum, and medical) in nature characteristic, the user can enter (pharmaceutical) and using the ontology it will be transformed to medical. Thereby, the providers will have more flexibility when designing their websites.

Conclusion

WCMS is a powerful tool for creating, editing, organizing, and publishing content. Even though WCMS might suggest a number of templates, but it does not consider the nature of the website to be developed. It has been noticed that websites that lack in its information and structure make the visitors get lost and feel disoriented. Thus, this manuscript proposed a new approach to improve WCMS using web personalization; such that WCMS can take advantage of previous developers' knowledge to offer different template designs for each website developer depending on their need. Such an approach merged web personalization with automatic template extraction to improve the quality of WCMS and increase its flexibility and scalability. Moreover, a new technique was developed, which mixes hybrid EA (GA and ACC) with CTM. At the end of the manuscript, an experiment has been conducted to conclude that the proposed approach has a high quality, in regard of speed, precision, and accuracy. It only takes few milliseconds to accurately match users and cluster the web pages. Finally, as future work, it has been recommended to use ontology to increase the approach scalability and flexibility.

REFERENCES

- Bille P (2005). A survey on tree edit distance and related problems. *Theor. Comput. Sci.* 337(1-3):217-239.
- Carminati B, Ferrari E, Perego A (2012). A multi-layer framework for personalized social tag-based applications. *Data & Knowl. Eng.* 79-80(0):62-86.
- Chávez F, Fernández F, Alcalá R, Alcalá-Fdez J, Olague G, Herrera F (2012). Hybrid laser pointer detection algorithm based on template matching and fuzzy rule-based systems for domotic control in real home environments. *Appl. Intell.* 36(2):407-423.
- Christos B, Vaggelis K, Ioannis M (2004). 5-7 April 2004). Web page fragmentation for personalized portal construction. Paper presented at the Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004.
- Chulyun K, Kyuseok S (2011). TEXT: Automatic Template Extraction from Heterogeneous Web Pages. *Knowledge and Data Engineering. IEEE Trans.* 23(4):612-626.
- CoreMedia (2012). CoreMedia Introduces New Web Content Management Solution for Media and Entertainment. 2012, from http://www.coremedia.com/web-content-management/news-events/press-releases/coremedia-launches-new-web-content-management-solution-for-media-and-entertainment-/1602/10830/_/11n0rb8z/-/index.html.
- Crossley M, Kings NJ, Scott JR (2003). Profiles-Analysis and Behaviour. *BT Technol. J.* 21(1):56-66.
- Deneubourg L, Goss S, Franks N, Detrain C, Chretien L (1991). *The Dynamics of Collective Sorting: Robot-Like Ant and Ant-Like Robot*. Paper presented at the from animals to animats: Proceedings of the first International Conference on Simulation of Adaptive Behavior pp. 183-192.
- Dinh T, Mamun A (2004). A combination of clustering algorithms with Ant Colony Optimization for large clustered Euclidean Travelling Salesman Problem. Paper presented at the WSEAS SOSM 2004, Miami, Florida.
- Eirinaki M, Mavroeidis D, Tsatsaronis G, Vazirgiannis M (2006). Introducing Semantics in Web Personalization: The Role of Ontologies. In M. Ackermann, Berendt B, Grobelnik M, Hotho A, Mladenec D, Semeraro G, Spiliopoulou M, Stumme G, Svátek V, van Someren M. (Eds.), *Semantics, Web and Mining* 4289:147-162 Springer Berlin / Heidelberg.
- Eirinaki M, Vazirgiannis M (2003). Web mining for web personalization. *ACM Trans. Internet Technol.* 3(1):1-27.
- ElGibreen H, El-Masri S (2012). Template Personalization and Evolutionary Algorithms. Paper presented at the GEM'12 - 9th International Conference on Genetic and Evolutionary Methods.
- Eslami MZ, Zarghami A, Sapkota B, Sinderen MV (2011). Personalized Service Creation by Non-technical Users in the Homecare Domain. *Proc. Comp. Sci.* 5(0):409-417.
- Ferrara E, Baumgartner R (2011). Automatic Wrapper Adaptation by Tree Edit Distance Matching. *CoRR*, abs/1103.1252.
- Flesca S, Greco S, Tagarelli A, Zumpano E (2004). Non-invasive support for personalized navigation of Websites. Paper presented at the International Proceedings of Database Engineering and Applications Symposium, 2004. IDEAS '04 pp. 183-192.
- Freitas A (2002). Data Mining and Knowledge Discovery with Evolutionary Algorithms. Berlin: Springer-Verlag. Pp. 1-260.
- Gui-Sheng Y, Guang-Dong G, Jing-Jing S (2010). A template-based method for theme information extraction from web pages. Paper presented at the International Conference on Computer Application and System Modeling (ICCSM), 2010. pp. V3-721-V3-725.
- Haikun H, Xiaoxin C, Guoshi W, Jing L (2010). Web Data Extraction Based on Tree Structure Analysis and Template Generation. Paper presented at the International Conference on E-Product E-Service and E-Entertainment (ICEEE), 2010. pp. 1-5.
- He L, Chen Y (2010). Design and implementation of Web Content Management System by J2EE-based three-tier architecture: Applying in maritime and shipping business. Paper presented at the 2nd IEEE International Conference on Information Management and Engineering (ICIME), 2010. pp. 513-517.
- Hildebrandt M (2008). Defining Profiling: A New Type of Knowledge? In M. Hildebrandt & S. Gutwirth (Eds.), *Profiling the European Citizen* Springer Netherlands, pp. 17-45.
- Ji X, Zeng J, Zhang S, Wu C (2010). Tag tree template for Web information and schema extraction. *Expert Syst. Appl.* 37(12):8492-8498.
- Kim Y, Park J, Kim T, Choi J (2007). Web Information Extraction by HTML Tree Edit Distance Matching. pp. 2455-2460.
- Laahs K, McKenna E, Vanamo VM (2008). 15 - Site Definitions and Templates Microsoft SharePoint 2007 Technologies, Burlington: Digital Press pp. 429-462.
- Lu J, Ruan D, Zhang G (2007). E-Service Intelligence: An Introduction. In Lu J, Zhang G, Ruan D (Eds.), *E-Service Intelligence*, Springer Berlin / Heidelberg 37:1-33.
- Munk M, Kapusta J, Švec P (2010). Data preprocessing evaluation for web log mining: Reconstruction of activities of a web visitor. *Proc. Comp. Sci.* 1(1):2273-2280.
- Nasraoui O, Soliman M, Saka E, Badia A, Germain R (2008). A Web Usage Mining Framework for Mining Evolving User Profiles in Dynamic Web Sites. *Knowledge and Data Engineering. IEEE Trans.* 20(2):202-215.
- net.sourceforge.htmlcleaner. (2010). HtmlCleaner release 2.2. 2011, from <http://htmlcleaner.sourceforge.net>.
- Perugini S (2010). Personalization by website transformation: Theory and practice. *Information Processing & Management* 46(3):284-294.
- Rana C (2012). A Study of Web Usage Mining Research Tools. *Adv. Netw. Appl.* 03(6):1422-1429.
- Ravi J, Yu Z, Shi W (2009). A survey on dynamic Web content generation and delivery techniques. *J. Network Comp. Appl.* 32(5):943-960.
- Russell S, Norvig P (2003). *Artificial Intelligence: A Modern Approach* (Second ed.): Prentice Hall.
- Schiaffino S, Amandi A (2009). Intelligent User Profiling. In M. Bramer (Ed.), *Artificial Intelligence - An International Perspective*: Springer Berlin / Heidelberg 5640:193-216.
- Shen Hz, Zhao Jd, Yang ZZ (2006). A Web Mining Model for Real-time Webpage Personalization. Paper presented at the International Conference on Management Science and Engineering, ICMSE '06.
- Shui-Lung C, Jane Yung-jen H (2004). Tree-Structured Template Generation for Web Pages. Paper presented at the International Conference on Web Intelligence, IEEE/WIC/ACM 2004.
- Wan Mohd Mahidin WAMA (2003). Dynamic template for lecturers' webpages in FTMSK. Universiti Teknologi MARA.
- Wei S, Jian-Chang L, Yu-Jun H, Jian-Qiang L (2005). Application of neural network model combining information entropy and ant colony clustering theory for short-term load forecasting. Paper presented at the International Conference on Machine Learning and Cybernetics, 2005.
- What is the Document Object Model? (2004). 2011, from <http://www.w3.org/TR/DOM-Level-3-Core/introduction.html>.
- Wool J, Woo S (2005). New Encoding/Converting Methods of Binary GA/Real-Coded GA. *IEICE Trans.* E88-A(6):1545-1156.
- Yen BPC, Kong RCW (2002). Personalization of information access for electronic catalogs on the web. *Elect. Commerce Res. Appl.* 1(1):20-40.