Personalized Hotlink Assignment using Social Networks

Christos Makris, Konstantinos Siaterlis and Pantelis Vikatos
Computer Engineering & Informatics Department, University of Patras, Patras, Greece

Keywords: Hotlink Assignment, Personalization, Social Networks.

Abstract: In this paper, we introduce a novel methodology for personalized website reconstruction. We combine context and popularity of the web pages and the information of user's interest from social media. We present an efficient automatic web restructure placing suitable hotlinks between nodes of the generated website's graph using information of social media contrary to previous studies. In addition, our methodology includes an innovative personalization scheme using a topic modeling approach to texts of users of social media to create a graph of categories. We evaluate our approach counting user's feedback about the ordering and relevance of links to a website.

1 INTRODUCTION

It is difficult to cover the needs of all users when the number of pages and categories in a dynamic website increase. Many websites contain hundreds or thousands of different categories, creating difficulties to find the page that the user wants degrading the quality of the website. Also, the amount of information that is provided by various agents grows rapidly according to informational needs. The reconstruction of a website based on the fluctuation of web page's popularity constitutes a significant factor to enhance websites. The goal of reconstruction is to reach popular pages in fewer steps starting from the homepage, improving the browsing experience. A well-cited methodology is the use of additional links i.e. hotlinks, that connect popular web pages with its descendants reducing the distance from the home page. A first approach has been introduced in (Perkowitz and Etzioni, 2000) and presents the idea of a modification of the link structure of the website, minimizing the steps from homepage to popular pages using hotlinks. However, the unilateral use of popularity as a factor of browsing enhancement might not be efficient due to the fact that different users have different needs based on their preferences. Therefore the introduction of personalization in the website reconstruction is necessary in order to provide more targeted information. The extraction of user's interest and needs occurs in an explicit or implicit manner. The main drawback of an explicit collection of user feedback that it is not supported by all users. In many studies, the implicit discovery of preferences is fed by the browsing history of a user e.g. clicks. An alternative approach is the use of the available information from social media in which a user participates. Our study paper examines a methodology of website's reconstruction by the concept of hotlink assignment from a new point of view. An algorithm is presented to ameliorate the accessibility of non-popular pages which however are highly conceptually relevant and recent trend according to social media, by adding extra links to them from highly popular pages, resulting to fewer hops from the homepage. Our methodology uses a local metric to recognize the accessibility of each web page separately and the target is its minimization. Our study also describes an innovative personalization scheme discovering a user's interest through social media.

The main points of our contribution can be summarized in the following sentences:

- We propose a holistic procedure of a website's reconstruction.
- A generic scheme describes personalization through social media.
- We formulate a personalized hotlink assignment algorithm.
- We evaluate our methodology through users' relevance feedback.

The rest of the paper is structured as follows. Section 2 overviews related work, we motivate our research from current challenges and related studies. In Section 3, we provide an overview of the methodology describing each independent task. It is noted we present the algorithm of personalized hotlink assign-

Makris, C., Siaterlis, K. and Vikatos, P.
Personalized Hotlink Assignment using Social Networks.
DOI: 10.5220/0006296800710079
In *Proceedings of the 13th International Conference on Web Information Systems and Technologies (WEBIST 2017)*, pages 71-79
ISBN: 978-989-758-246-2

ment in Section 3.4.2. Section 4 provides an overview of the implementation of the system for modules and sub-modules respectively and presents a reference to our experimental results. Finally, in Section 5, we discuss the strengths and limitations of our approach and we conclude the paper with an outlook to future work.

2 RELATED WORK

Enhancing web browsing experience has gained the interest of researchers. The concept of assigning hotlinks to websites has been suggested by Perkowitz and Etzoni in (Perkowitz and Etzioni, 2000) where a site is transformed using shortcutting in order to be browsed efficiently by users. The clairvoyant user model (Bose et al., 2000; Czyzowicz et al., 2001; Kranakis et al., 2001) and the greedy user model (Gerstel et al., 2003; Jacobs, 2010; Jacobs, 2011; Matichin and Peleg, 2007; Pessoa et al., 2004a; Pessoa et al., 2004b) constitute the main methodologies in which the presence of the hotlinks is known only for the present node and the whole site respectively. Based on the clairvoyant model to study (Bose et al., 2000) has presented the problem of assigning hotlinks proving that solving the problem to a directed acyclic graph (DAG) is NP-hard and introducing the upper and lower bounds on the expected number to reach leaves from the root (homepage) of a complete binary tree. Considering the website as a tree, study (Czyzowicz et al., 2001) shows an $O(n^2)$ algorithm for assigning a hotlink which outperforms greedy approaches. Studies (Gerstel et al., 2003; Pessoa et al., 2004a) are focused in the greedy model with running time exponential in the depth of the tree and thus polynomial for trees of logarithmic depth and an implementation of this algorithm able to discover optimal solutions for trees as presented in (Pessoa et al., 2004b). An approach of the natural greedy strategy achieves at least half of the gain of an optimal solution. An algorithm of 2-approximation in terms of the gain has been presented in (Matichin and Peleg, 2007). An improvement to hotlink assignment is presented in (Douïeb and Langerman, 2005) in which a linear-time algorithm where dynamic operations such as node insertion, deletion and weight reassignment are available. A common feature of all these studies is that they do not combine the popularity of pages with information on social media contrary to our study. Another difference of our approach is that we handle website as a directed acyclic graph contrary to the assumption that the underlying model is a tree. Also, we provide a unique personalized reconstructed site for each user based on hotlink assignment.

Also, the current scientific interests focus on personalization schemes for search engines, web pages and information systems. Personalization methodologies can be summarized in three main categories in the way that the necessary information is collected i.e. Explicit, implicit and hybrid. A plethora of web pages and information systems use explicit personalization and infer user interest based on predefined categories that the user should select. For example, the well-known search engine Google asks users to create a profile by selecting categories of interests. Contrary to this technique study (Kelly and Teevan, 2003) examines the improvement to search accuracy through personalization using implicit feedback information. Another proposed method (Matthijs and Radlinski, 2011) collects web usage data e.g. page session, URL and duration of visit; to discover the preferences of a user. Also, study (Peng et al., 2012) proposes a structure of categories (tree) with reference to Google directory. The category tree is updated through visits of websites and shows the degree of interest. Also, there are studies that combine implicit feedback from user and information of social media interactions. Study (Carmel et al., 2009) describes the creation of a profile for each user using social networks for improvement in web search. Also, study (Zhou et al., 2012) gathers information from social media and implicitly personalizes the search results via query expansion. Furthermore, hybrid personalization schemes combine implicit and explicit methods as study (Noll and Meinel, 2007) presents. Our methodology differs from the previous ones by using a topic modeling algorithm to social media text in order to create a graph where each node constitutes a category of interest and each weighted link the correlation between categories. The importance of nodes in terms of ingoing and outgoing links describes users' preferences.

3 MODEL OVERVIEW

A summary of independent tasks that our methodology consists of is given below:

- *Generation of website's graph.* The website is modeled as a directed graph $G(V,E)$ where nodes and edges are pages and links of the website respectively. Each node has a popularity attribute and each edge includes a context-similarity weight.
- *Extraction of website's categories.* Pages include semantic information for search engine optimization purposes. We extract the category of each web page and create a list of categories for the whole website.

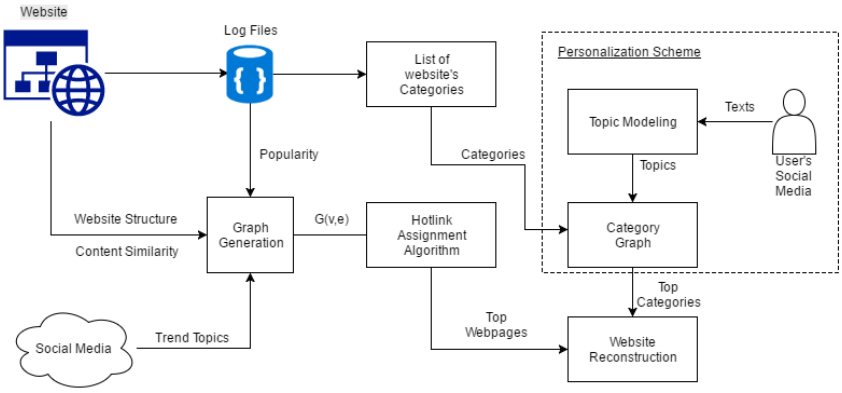

Figure 1: System Architecture.

- *Extraction of user's interest*. Our personalization scheme produces a list of categories derived from user's texts in social media. We gather raw texts that the user posts on social media, then a topic modeling algorithm produces the topics that are introduced to a personalized category graph and the Pagerank algorithm creates a ranked list of categories indicating user's top preferences.
- *Creation of Hotlinks*. We use shortcuts for a node to one of its descendants. The goal is the enhancement of the browsing experience. The distance between the home page and popular pages is reduced.
- *Reconstruction of website*. A variation of Pagerank algorithm prioritizes pages in terms of popularity, context-similarity and personal interest. The existence of new links favors the reconstruction of the website.

In the following subsection tasks and modules of our model are described in detail and Figure 1 presents the system architecture.

3.1 Generation of Website's Graph

We model the website as a graph $G(V,E)$. Our procedure uses crawling which is handled by a dedicated crawler that is developed for the task and that allows sampling pages and links in a manner that network properties are preserved and can be used in our modeling procedure. The crawling is oriented to discovering new nodes in a Breadth-First search (BFS) approach. We introduce an attribute calculating the popularity of each page/node and taking account a page's clicks as well as ranking in the social media trend list. In addition, we use a term based text similarity approach to compare all web page pairs. The value of similarity in each pair is stored in NxN matrix where N is the number of different web pages.

3.2 Extraction of Categories

In our methodology, we declare that each page belongs to one category. For instance, the bbc.com uses categories such as news, sports, TV, music to structure the information. Initially, the task finds the category of each page using 2 different approaches:

1. Our crawler isolates the semantic information of the page in order to extract the category of the page e.g. RDF/XML
2. Topic modeling is performed at the text a web page contains for the extraction of the topics. From those topics a category is assigned.

We gather the categories and create a list which is used in our personalization scheme as it is described below.

3.3 Personalization Scheme

In this section, our personalization approach is described. There are three main approaches to extract a user's interests for personalization purposes i.e. Explicit, implicit & hybrid. Our scheme belongs to the implicit category using information through social media in order to determine the user's preferences as Figure 2 presents. Our personalization scheme has been inspired from study (Makris et al., 2008) in which a graph of categories is used depicting the current user's preferences in search engine results in a query. We adopt the method of using a graph, but we differentiate extracting topics from the social network. Our methodology extracts this type of information via social media and more specifically using user's texts e.g. posts, tweets. The rest of this section is dedicated to explaining in depth the methodology of our scheme.
First of all, the system can operate as intended given the fact that:

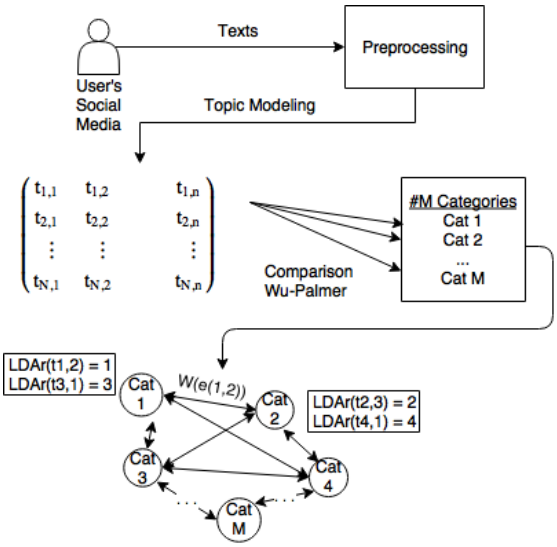

Figure 2: Personalization Scheme.

- The user is active on social networks
- The user willingly connects his/her social network account with the system

A time window is defined and during this period texts are collected constituting the current user's interest. The next step is the creation of a supertext for each user which is refined by a preprocessing module including stop-words removal, tokenization and stemming. The outcome of the preprocessing step is a vector of words that is introduced in a Topic modeling algorithm. Latent Dirichlet Allocation (LDA) (Krestel et al., 2009) is used for the extraction of topics. The extracted topics are semantically compared with fixed categories. This step is necessary to connect the user's interests with the categories from an information system; e.g. search machine, website. To compare a topic with a category, Wu&Palmer metric (Equation 1) calculates semantic similarity using the depths of two synsets in the WordNet taxonomies (Wu and Palmer, 1994; Pedersen et al., 2004), along with the depth of the LCS (Least Common Subsumer).

$$wup(s1,s2) = \frac{2 * depth(LCS)}{depth(s1) + depth(s2)} \quad (1)$$

Each word in the extracted topics is compared to categories and the category with the maximum similarity is stored in a graph H if it exceeds a threshold T. Graph H called the category graph, is a complete graph containing nodes, which are the categories and weighted links between categories whose weights are formulated by the Equation 2,

$$w(e(u,v)) = \sum_k \sum_l (LDAr(t_k)[LDAr(t_l) - LDAr(t_k)])^{-1} \quad (2)$$

where t_k is the topic k that exceeds threshold T and belongs to the category u. $LDAr(t_k)$ and $LDAr(t_l)$ constitute the ranking of topic k and l by using LDA procedure respectively. All topic combinations of the nodes u, v are used.

For instance, to calculate the weight of the edge $e(1,2)$ in Figure 2, the ranking of topics Cat 1 and Cat 2 is extracted and the weight is calculated in the following manner: $w(e(1,2)) = \frac{1}{1*(2-1))} + \frac{1}{1*(4-1)} + \frac{1}{3*(2-3)} + \frac{1}{3*(4-3)} = 1 + \frac{1}{3} - \frac{1}{3} + \frac{1}{3} = 4/3$.

The weights in links are updated when a new sample of texts is mined for a specific user. The proposed personalization scheme is generic and can be applied to search engines in order to improve the ranking results.

3.4 Creation of Hotlinks

3.4.1 Hotlink Assignment

The concept of hotlink assignment (Perkowitz and Etzioni, 2000; Czyzowicz et al., 2001; Bose et al., 2000; Pessoa et al., 2004a) constitutes a methodology of websites' reconstruction, concerning the popularity of the web pages. The goal is the enhancement of browsing experience, reducing the distance between the homepage and popular nodes by adding hotlinks (shortcuts from a node to one of its descendants). According to previous studies, a website can be modeled as a tree $T = (V,E)$ which V is the set of web pages and E is the set of links. Each leaf-web page contains a weight representing the popularity of the web page. We declare that T^A is the tree formulated by an assignment A of hotlinks. The expected number of steps needed from the homepage to reach a web page on a leaf is calculated by Equation 3

$$E[T^A, pop] = \sum_{i\_is\_a\_leaf} d_A(i) pop(i) \quad (3)$$

where $d_A(i)$ is the distance of the leaf i from the root in T^A, and $pop = \{pop_i : i\_is\_a\_leaf\}$ is the probability distribution which derives from the distribution of popularity weights on the leaves in initial tree T. The minimization of this equation is the scope of the hotlink assignment algorithm.

Study (Antoniou et al., 2010) proposes an innovative methodology which not only uses the frequency of accessibility of popular pages, but also introduces the context-similarity of websites in the decision of adding extra links from non-popular pages. We use the same methodology as it is described in (Antoniou et al., 2010), however, our approach combines information given by social media in order to recalculate the value of popularity in each node. In our methodology a website is modelled as a directed acyclic graph

(DAG), $G(V,E)$ where $|V| = n$, i.e. website has n pages, and each edge has a weight $w_e(i,j) \in [0,1]$, which declares the content similarity between pages i and j. Also each node has a popularity weight pop_i. In related works (Perkowitz and Etzioni, 2000; Czyzowicz et al., 2001; Bose et al., 2000; Pessoa et al., 2004a; Antoniou et al., 2010), pop_i is calculated by the distribution of *clicks* in order to split web pages in POP and NonPOP sets which include the popular and non-popular web pages respectively. Contrary to this concept, we consider web pages that have many visits in the past may also be obsolete and outdated and thus we introduce the information of trending topics through social media.

Let $TT = \{tt_1, ..., tt_n\}$ be a set of Top-N trend topics from social media, ranked from 1 to n. Also C_i is the category of the i^{th} web page. The popularity of the web page is formed by social media current trends as the following equation shows:

$$rankFact = (1 + log_2(N - rank(tt_j)))$$
$$simFact = (1 + wup(tt_j, C_i))$$
$$pop_i = rankFact * simFact * clicks_i \quad (4)$$

where $wup(tt_j, C_i)$ is the maximum Wu&Palmer similarity (Wu and Palmer, 1994; Pedersen et al., 2004) between each trend and the category i, $clicks_i$ is the $i's$ web page's clicks and the $rank(tt_j)$ is the rank of the topic with the maximum Wu&Palmer similarity. The popularity of a website is increased if and only if the category has the same or nearby semantic with the trend and the trend is highly ranked. The lowest value of pop_i equals the number of visits to the web page ($clicks_i$).

The algorithm of hotlink assignment discovers all paths between a random page (source) of the graph and a page from the POP set (target). We count the popularity of a path as it follows:

$$Path\_pop_{i,j} = \sum_k \frac{pop_i}{\#in\_edges_i} \quad (5)$$

Figure 3 describes the calculation of the path's popularity. Candidate hotlinks are assigned between NonPOP nodes, that exist in the path with the maximum Path_pop, and the target node. There are two criteria that define the final selection of hotlinks. Firstly, we examine if the distance between the target and source is reduced. Then, we check the semantic similarity between target and NonPOP page. The hotlink with the most semantic similarity is created. The NonPOP page that is linked with a hotlink is removed from the NonPOP set and the procedure is continued until NonPOP set is empty. Then the remaining POP set is split as follows $|POP| = |NonPOP| = |Web - pages|/2$. The algorithm is described in detail in Section 3.4.2.

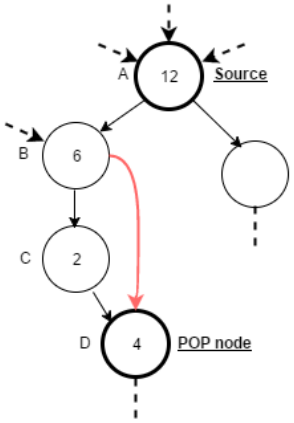

Figure 3: For example, $path\_pop(A,D) = \sum_i \frac{pop_{node_i}}{\#in\_edges_i} = 12/3 + 6/2 + 2/1 + 4/1 = 13$. Let (A,B,C,D) be the path with the maximum popularity. Let B be a node from NonPOP set, creating an edge(red line) the path is reduced by one.

3.4.2 The Algorithm

In this section, the personalized hotlink assignment is described in the Algorithm 1. The algorithm is initiated with the input elements of the Table 1.

Table 1: Algorithm's Variables.

Variable	Notion
G(V,E)	Website Structure as a graph
userID	User's social media id := j
WC	Website's Categories
Th	Threshold
CM	Context-Similarity Matrix
ST_jpr	Preprocessed super text of user j
C_j	Categories of Personalization Graph
TR	Social media trends
TP_j	List of topics of User j
$PG_j(V,E)$	Personalization Graph

At lines 1 through 5, the algorithm performs the initialization step and an ordered list of interest is extracted. The Procedure 1, $rank\_Cat\_Graph()$, is used to extract the ordered list of categories PR_j of each user j. Internally the procedure uses a preprocessing phase, which includes tokenization, the removal of stopwords, the extraction of tokens' lemma, in order to create a supertext of terms ST_jpr that they will be introduced in LDA algorithm. The PageRank algorithm creates the ranking list PR_j of categories in the category graph PG which each node is the website's category WC. The specification of nodes and edges' weight has been described in Section 3.3. At lines 6

to 9, the popularity of each node is calculated using Equation 4. At lines 10-16, the POP and NonPOP lists are created based on the distribution of popularity in the nodes of the graph G.

Procedure 1: rank_Cat_Graph().

1: **input** Th, WC, C_j, ST_jpr
2: **output** PR_j
3: $PG_j(V,E) = \{\}$
4: $TP_j = LDA(ST_jpr)$
5: **for each** $t \in TP_j$ **do**
6: maxSim, cat := findMax($wup(t,WC)$)
7: **if** $maxSim > Th$ **then**
8: $C_j(cat) = C_j(cat) \cup LDAr(t)$
9: **end if**
10: **end for**
11: **for each** $l \in C_j$ **do**
12: **for each** $m \in C_j$ **do**
13: $l, m := t_k$ topic on cat C_j
14: **if** $l \neq m$ **then**
15: $V_{PG_j} = V_{PG_j} \cup m \cup l$
16: $E_{PG_j} = E_{PG_j} \cup e(m,l)$
17: $w(e(m,l))$ = Equation 2
18: **end if**
19: **end for**
20: **end for**
21: **return** $PageRank(PG_j)$

From line 17 to 32, the algorithm creates the hotlinks as it is described in Section 3.4.1. A slightly adjusted PageRank (*Aux*) is used to rank the nodes of the graph H combining the incoming links with nodes' context-similarity nodes. The similarity between web pages is called *Simetric* and is initially equal to $\frac{1}{\#webpages}$ and *Aux* is equal to 0. We use the following recursive formula for web page i:

$$Aux(i) = Aux(i) + \frac{\sum_{v \in In(i)} Simetric(v) * sim(v,i)}{\#outgoing\_links\_of\_v} \quad (6)$$

where $In(i)$ is the incoming web pages to page i, $Simetric(i) = \frac{q}{\#webpages} + (1-q) * Aux(i)$ where q is a dumping factor and $sim(v,i)$ is the context-similarity of web pages v and i.
The output of the algorithm is a ranking list of web pages based on popularity, context-similarity and personalized information from social media.

3.5 Reconstruction of Website

Algorithm 1 provides a list of links in a descending order. Our approach uses this information in order to reconstruct the website.

Algorithm 1: Personalized Hotlinks.

1: **input** $G(V,E)$, userID, WC, Th, $T_{i,j}$, C_j, TR, ST_jpr
2: **output** $H(V,E)$, $R_j = \{\}$
3: $H = G$
4: PR_j = **rank_Cat_Graph**($Th, WC, C_j, PG_j(V,E), ST_jpr$)
5: $G_{new} = G$
6: **for each** $node \in G_{new}$ **do**
7: $maxSim$, $rank(tt_j)$ = findMax($wup(TR, node\{'category'\})$)
8: w_{node} = Equation 4
9: **end for**
10: **while** $|PoP| < |G|/2$ **do**
11: $maxPOP$, $maxNode$ = findMaxWeight(G_{new})
12: $G_{new} = G_{new} - \{maxNode\}$
13: $POP = POP \cup maxNode$
14: **end while**
15: $NonPOP = NonPOP \cup V_{G_{new}}$
16: $P_{POP} = probabilityDistr(POP)$
17: **while** $NonPOP \neq \{\}$ **do**
18: $source$ = chooseRand(G,$1/|G|$)
19: $target$ = chooseRand(POP,P_{POP})
20: $Path$ = findMaxPopPath(G, $source$, $target$)
21: $Path = Path - \{Path \cap POP\}$
22: **for each** $node \in Path$ **do**
23: $G_{temp} = H$
24: $E_{G_{temp}} = E_{G_{temp}} \cup e(target, node)$
25: **if** minPath(G_{temp}, $source$, $target$) < minPath(G, $source$, $target$) **then**
26: $candNodes = candNodes \cup node$
27: **end if**
28: **end for**
29: $y = findMaxCSim(candNodes, target)$
30: $E_H = E_H \cup e(target, y)$
31: **end while**
32: $R = Aux(H)$

We declare that this procedure does not remove links or nodes from the website. The reconstructed website is updated when a current time window of social media crawling occurs or new pages are added to the website. We denote that changes of user's interest can be detected by continuous sampling of his/her social media activity. Also, we introduce the information of trending topics through social media in order to filter web pages that, even if they had many visits in the past might be obsolete. The restructure starts from the homepage of the website. The context of the homepage and the links to other pages remain on the page. We retrieve from graph H the links of the page and the new ones (hotlinks) that have been created. We place the new links based on the ranking list. We

traverse the graph using the outgoing links in a Breadth-First search (BFS) approach. The procedure continues until the content of all pages and the new links are placed on the reconstructed website.

4 EXPERIMENTAL RESULTS

4.1 Implementation

We conducted an experimental procedure using relevance feedback from users. The scenario includes the creation of a web interface in which users are registered via a Twitter account. If the Twitter account is active our system gathers the necessary information (tweets & tweets of user's friends). Then, a ranking list of links with title, description and image is presented. The ranking list is produced using the algorithm of hotlink assignment as it is presented in (Antoniou et al., 2010). Each user has the ability to read the description, browse to the provided links and the obligation to score each page in scale 0-3 according to his/her interest, where 0 means not interested and 3 means very interested. After submission of scores, a new list of ranked links is presented based on Algorithm 1. The user acts in the same way as in the previous stage.

We aggregated news and their links from the BBC[1] and a graph was created based on this well-known website. The popularity of each page was retrieved from the web-based company Alexa[2] which provides popularity metrics for domains and subdomains.

We used Twitter to extract the necessary information. Our server gathers the recent 100 tweets from the tested user as well as tweets from the recently mentioned user's friends. The selected tweets initiate the personalization procedure as described in Section 3.3. It is noted that the LDA algorithm is formulated for 10 topics in user's supertext. Based on tests that we selected the average number of tweets and mentioned friends was 73 and 34 respectively and the average number of nodes in the category graph was 24 as Table 3 shows.

Our system was implemented in Python 2.7. We collected data from users and the users' friends using the Twitter API and we performed topic modeling on the tweets using LDA[3]. We used NLTK[4] module for preprocessing and content similarity measurement and networkx[5] module for graph handling. The web interface for the evaluation phase was designed with PHP/HTML and users' relevance feedback data was stored on a dedicated MySQL server for post analysis.

4.2 Results

Providing a better insight into the quality of our experimental data, Table 2 and Table 3 present the statistics of the website's DAG as well as the average tweets per user and the number of friends we have sampled respectively.

Table 2: Hotlink Assignment Graph Stats.

Properties	Value
Nodes	1220
Edges	2226
Avg Degree	1.824
Longest Path	15

Table 3: Personalization Stats.

Properties	Value
Avg #tweets	73.6
Avg # rel categories	24.3
Avg # of friends	34.2

In Figure 4 we compare the distances from homepage to random nodes between the context-similarity based hotlinks assignment as it is presented in study (Antoniou et al., 2010) and our implementation. We present the percentage improvement in the distance as we gradually increase the size of the website's DAG. We can consider that our implementation outperforms and improves the distance by an average of 2%. Furthermore, Figure 5 presents the improvement of ranking on the nodes between our implementation and the algorithm of (Antoniou et al., 2010). We present the number of improved nodes increasing gradually with the size of the website's DAG. We can see that the ranking of nodes is improved as we raise the Aux of nodes relevant to the trending topics from social media. Our website consists of 1200 nodes and the algorithm ranks 60 nodes better due to trending topics. Furthermore, we evaluate users' relevance feedback via the normalized Discounted Cumulative Gain (nDCG) as Equation 7 shows

$$nDCG_p = \frac{DCG_p}{iDCG_p} \tag{7}$$

[1] http://www.bbc.com/news
[2] http://www.alexa.com/
[3] https://pypi.python.org/pypi/lda
[4] http://www.nltk.org/
[5] https://networkx.github.io

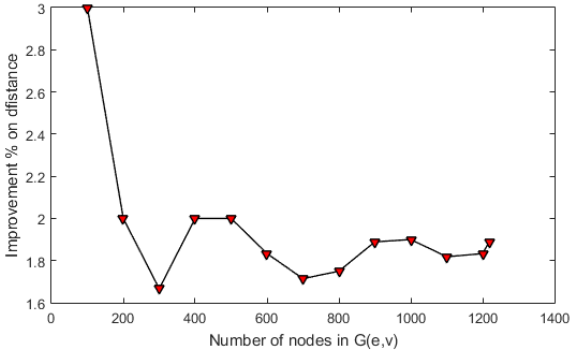

Figure 4: Difference % on improved distance.

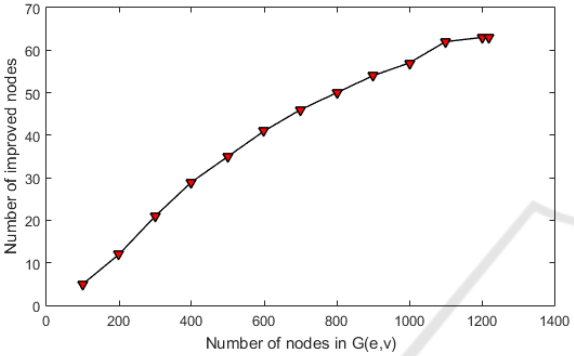

Figure 5: Improvement on nodes' ranking.

$$DCG_p = rel_1 + \sum_{i=1}^{p} \frac{rel_i}{log_2(i+1)} \quad (8)$$

where, rel_i is user's response in the range 0(not interested) - 3(very interested) for the i_{th} result. The iDCG is the ideal ranking based on user's preferences. We present the nDCG values in Figure 6. We conducted the experiment over 30 individuals[6] . Each individual evaluates our methodology for 10 independent times. We present the average nDCG value for each individual and for both implementations in Figure 6. We consider that our implementation efficiently targets the user's interests in comparison to study (Antoniou et al., 2010). More specifically 83% of users' responses declare the provided information on our methodology depicts their preferences.

5 CONCLUSIONS

Our study deals with the problem of personalization in hotlink assignment. The first innovation in our methodology is the use of information about social media and, in particular, trend topics on Twitter in order to recalculate the attribute of popularity in each

[6]undergraduate students of Computer Engineering and Informatics Department, University of Patras, Greece

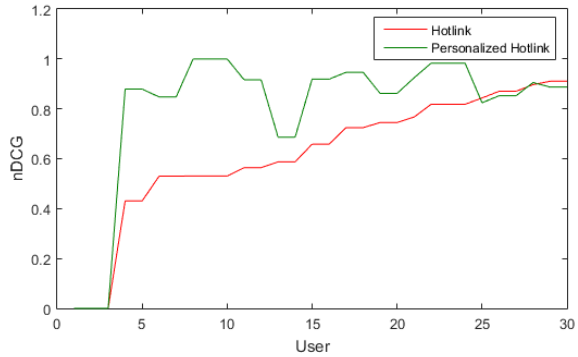

Figure 6: nDCG - Hotlink vs Personalized Hotlink.

node of the graph. It is noted that previous works deal with the popularity of a web page only with clicks that pages receive from users during browsing. The second contribution is the detailed description of a personalization scheme that handles explicit information from raw texts of social media and creates a ranking list of categories describing users' interests. The scheme is generic and can be used in implementations that need personalization such as search engines and information systems. Also, a new algorithm of personalized hotlink assignment is described. According to the experimental procedure, our methodology provides efficient results in terms of distance between the homepage and other pages, ranking of web pages and relevance on users' preferences.

The main points of our contribution based on the experimental results can be summarized in the following sentences:

- We reduce the distance between the homepage and other popular pages in the graph in a way that affects the browsing experience due to the fact that users can reach in fewer steps his/her preferences.
- We differentiate the nodes ranking via relevance of the trend topics on Twitter. The experiments show that the increase of the number of nodes is correlated to the number of nodes that are improved in terms of ranking.
- We provide a ranking list of web pages that efficiently targets the user's interests.

As future work, we are interested in examining the comparison of our methodology in different social networks such as Facebook and identify the parameters that influence the results of our algorithm. Our study can capture user's preferences in a single language, however. our plan is to extend our work for multilingual personalization.

REFERENCES

Antoniou, D., Garofalakis, J., Makris, C., Panagis, Y., and Sakkopoulos, E. (2010). Context-similarity based hotlinks assignment: Model, metrics and algorithm. *Data & Knowledge Engineering*, 69(4):357–370.

Bharambe, I. and Makhijani, R. K. (2014). Design and implementation of search engine using vector space model for personalized search. *International Journal of Advanced Research in Computer Science and Software Engineering, ISSN*, 2277:1019–1023.

Bose, P., Czyzowicz, J., Gasieniec, L., Kranakis, E., Krizanc, D., Pelc, A., and Martin, M. V. (2000). Strategies for hotlink assignments. In *International Symposium on Algorithms and Computation*, pages 23–34. Springer.

Carmel, D., Zwerdling, N., Guy, I., Ofek-Koifman, S., Har'El, N., Ronen, I., Uziel, E., Yogev, S., and Chernov, S. (2009). Personalized social search based on the user's social network. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 1227–1236. ACM.

Czyzowicz, J., Kranakis, E., Krizanc, D., Pelc, A., and Martin, M. V. (Jun. 25-28, 2001). Evaluation of hotlink assignment heuristics for improving web access. In *Proceedings of the Second International Conference on Internet Computing (IC'01)*, volume 2, pages 793–799. Citeseer.

Douïeb, K. and Langerman, S. (2005). Dynamic hotlinks. In *Workshop on Algorithms and Data Structures*, pages 182–194. Springer.

Gerstel, O., Kutten, S., Matichin, R., and Peleg, D. (2003). Hotlink enhancement algorithms for web directories. In *International Symposium on Algorithms and Computation*, pages 68–77. Springer.

Jacobs, T. (2010). An experimental study of recent hotlink assignment algorithms. *Journal of Experimental Algorithmics (JEA)*, 15:1–1.

Jacobs, T. (2011). Constant factor approximations for the hotlink assignment problem. *ACM Transactions on Algorithms (TALG)*, 7(2):16.

Kelly, D. and Teevan, J. (2003). Implicit feedback for inferring user preference: a bibliography. In *ACM SIGIR Forum*, volume 37, pages 18–28. ACM.

Kranakis, E., Krizanc, D., and Shende, S. (2001). Approximate hotlink assignment. In *International Symposium on Algorithms and Computation*, pages 756–767. Springer.

Krestel, R., Fankhauser, P., and Nejdl, W. (2009). Latent dirichlet allocation for tag recommendation. In *Proceedings of the third ACM conference on Recommender systems*, pages 61–68. ACM.

Makris, C., Panagis, Y., Plegas, Y., and Sakkopoulos, E. (2008). An integrated web system to facilitate personalized web searching algorithms. In *Proceedings of the 2008 ACM symposium on Applied computing*, pages 2397–2402. ACM.

Matichin, R. and Peleg, D. (2007). Approximation algorithm for hotlink assignment in the greedy model. *Theoretical Computer Science*, 383(1):102–110.

Matthijs, N. and Radlinski, F. (2011). Personalizing web search using long term browsing history. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 25–34. ACM.

Noll, M. G. and Meinel, C. (2007). Web search personalization via social bookmarking and tagging. In *The semantic web*, pages 367–380. Springer.

Pedersen, T., Patwardhan, S., and Michelizzi, J. (2004). Wordnet:: Similarity: measuring the relatedness of concepts. In *Demonstration papers at HLT-NAACL 2004*, pages 38–41. Association for Computational Linguistics.

Peng, X., Niu, Z., Huang, S., and Zhao, Y. (2012). Personalized web search using clickthrough data and web page rating. *Journal of Computers*, 7(10):2578–2584.

Perkowitz, M. and Etzioni, O. (2000). Towards adaptive web sites: Conceptual framework and case study. *Artificial intelligence*, 118(1):245–275.

Pessoa, A. A., Laber, E. S., and de Souza, C. (2004a). Efficient algorithms for the hotlink assignment problem: The worst case search. In *International Symposium on Algorithms and Computation*, pages 778–792. Springer.

Pessoa, A. A., Laber, E. S., and de Souza, C. (2004b). Efficient implementation of hotlink assignment algorithm for web sites. In *In Proceedings of the Workshop on Algorithm Engineering and Experiments*, pages 79–87.

Wu, Z. and Palmer, M. (1994). Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 133–138. Association for Computational Linguistics.

Zhou, D., Lawless, S., and Wade, V. (2012). Improving search via personalized query expansion using social media. *Information retrieval*, 15(3-4):218–242.