Exploiting Linguistic Analysis on URLs for Recommending Web Pages: A Comparative Study

Sara Cadegnani[1], Francesco Guerra[1(✉)], Sergio Ilarri[2], María del Carmen Rodríguez-Hernández[2], Raquel Trillo-Lado[2], Yannis Velegrakis[3], and Raquel Amaro[4]

[1] Università di Modena e Reggio Emilia, Modena, Italy
{sara.cadegnani,francesco.guerra}@unimore.it
[2] University of Zaragoza, Zaragoza, Spain
{silarri,raqueltl}@unizar.es, mary0485@gmail.com
[3] University of Trento, Trento, Italy
velgias@disi.unitn.eu
[4] Universidade Nova de Lisboa, Lisbon, Portugal
raquelamaro@fcsh.unl.pt

Abstract. Nowadays, citizens require high level quality information from public institutions in order to guarantee their transparency. Institutional websites of governmental and public bodies must publish and keep updated a large amount of information stored in thousands of web pages in order to satisfy the demands of their users. Due to the amount of information, the "search form", which is typically available in most such websites, is proven limited to support the users, since it requires them to explicitly express their information needs through keywords. The sites are also affected by the so-called "long tail" phenomenon, a phenomenon that is typically observed in e-commerce portals. The phenomenon is the one in which not all the pages are considered highly important and as a consequence, users searching for information located in pages that are not condiered important are having a hard time locating these pages.

The development of a recommender system than can guess the next best page that a user wouild like to see in the web site has gained a lot of attention. Complex models and approaches have been proposed for recommending web pages to individual users. These approached typically require personal preferences and other kinds of user information in order to make successful predictions.

In this paper, we analyze and compare three different approaches to leverage information embedded in the structure of web sites and the logs of their web servers to improve the effectiveness of web page recommendation. Our proposals exploit the context of the users' navigations, i.e., their current sessions when surfing a specific web site. These approaches do not require either information about the personal preferences of the users to be stored and processed, or complex structures to be created and maintained. They can be easily incorporated to current large websites to facilitate the users' navigation experience. Last but not least, the paper reports some comparative experiments using a real-world website to analyze the performance of the proposed approaches.

N.T. Nguyen et al. (Eds.): TCCI XXVI, LNCS 10190, pp. 26–45, 2017.
DOI: 10.1007/978-3-319-59268-8_2

1 Introduction

A great deal of of websites, in particular websites of Public Administration institutions and govermental bodies, contain a large amount of pages with a lots of information. The content of this kind of websites is usually very wide and diverse, as it is targeting a broad group of diverse users. Moreover, these institutions are frequently the owners and the reference authorities of most of the content offered in their web pages (i.e., they are not simple information aggregators, but they are the providers of authoritative information). Therefore, a huge amount of visitors is interested in exploring and analyzing the information published on them. As an example, the ec.europa.eu and europa.eu websites, managed by the European Commission, have been visited by more than $520M$ people in the last year[1].

The websites of these institutions offering large amounts of data are typically organized in different thematic categories and nested sections that generally form large trees with a high height (e.g., the previously cited website is organized in six sections: "The Commission and its Priorities", "About the European Commission", "Life, work and travel in the EU", etc.). Nevertheless, users usually consider the retrieval of useful information a difficult task since the way in which the information is organized (i.e., the conceptualization of the website) can differ from what they expect when they are surfacing it, and unfrequent information demands usually require them to spend a lot of time in order to locate the information they need. So, some techniques and best practices for the design these websites have been proposed and experimented along the time. In some websites, for example, the information is grouped according to the topic. In other websites, a small set of different profiles (types of users) are defined and users are explicitly asked to choose one of those roles to surface the websites (e.g., in a university website, users can be asked to declare if they are students, faculty members, or companies, and according to this and the information provided when they enter in sections of the website, the information is structured in different ways). However, the "long tail" phenomenon[2] also affects the task of searching information in this kind of websites, where there are thousands of pages that can be accessed any time, independently of their publication date.

Different approaches and techniques have been proposed to improve users' experience navigating large websites. One of the solutions typically adopted is to include a search form in the header of the web pages to allow users to express their information needs by mean of keyword queries. Another approach to support users is to provide users with a little frame or area in the web page (or a special web page) where a list of "suggested links" is shown. The main disadvantage of the first approach is that it requires to maintain updated a complex indexed structure which must change when the web pages are modified (additions, removals, updates of content, etc.). Even if the data to search is stored in a

[1] http://ec.europa.eu/ipg/services/statistics/performance_en.htm, statistics computed on June 1st, 2015.

[2] http://www.wired.com/2004/10/tail/.

structured database, the issue remains since keyword queries against databases are not easily solvable [3,4]. Besides, it requires that users explicit through keywords their information needs, which could be difficult for some users. Moreover, there exists a semantic gap between the users' information needs and the queries submitted to the search system. With respect to the second option, two trends have been identified: (1) showing the same content to all the users visiting the website at a specific moment, and (2) considering the profile of each user to offer him/her a personalized list of suggested links. Showing all users the same recommendations cannot be appropriate, as this type of websites are oriented to a wide heterogeneous public, and what is interesting for a visitor can be useless for another. On the other hand, maintaining profiles of users implies that the users should be registered in the website, and profiled with respect their interest. This also leads to the need (1) to take into account complex and reliable procedures to securely maintain their personal information while respecting their privacy and legal issues, and (2) to effectively profile the users on the basis of the (few) personal data available.

In this paper, we analyze and compare three different approaches to create a dynamic list of "suggested links to web pages of the website" which consider information embedded in the structure of the website and the logs of their web servers. In particular, our proposals for recommender systems take into account:

- *The web pages that the user is visiting in the current session.* The recommendation system works in real time and dynamically updates the links to propose by taking into account the pages he/she is navigating. Moreover, the suggested links are updated after new pages are visited in a specific session.
- *Navigational paths (routes) of previous users.* By analyzing the logs of the web servers of the website, we can discover the next pages visited by other users when they were in the same page as the current user. In particular, we consider that the users' navigation "sessions" extracted from the logs are sets of pages related to each other that satisfy the same information need. In fact, we assume that in a session the user is looking for something to satisfy a specific information need and that the session contains all the pages required for satisfying that need. In this way, the historical sessions can play the role of "suggestion spaces", as they include pages considered relevant in the same "context".
- *The website structure.* The structure of a website follows a conceptual taxonomy that is exploited for the recommendation, by suggesting more specific or more general web pages than the current one.
- *Lexical and semantic knowledge about the pages.* The content of the pages is used in order to suggest pages with a similar content. The extraction of keywords/topics representing the content can be a huge and complex task for some websites. For this reason, we tried to exploit the URL as a means for approximating the content of the pages. This idea is based on the observation that in some particular websites the URLs are highly explicative in the sense that they contain a lot of textual information about the pages and the categories the pages belong to. If this is the case for the website under analysis, we

can exploit this information in order to make suggestions. It should be noted that the use of descriptive URLs is a usual recommendation for SEO (Search Engine Optimization); moreover, thanks to the use of descriptive URLs, end users can anticipate what they can expect from a web page.

In this paper (extended version, with new experiments and discussion, of [5]), we analyze and compare three methods to make the recommendations: (1) No History method (NoHi), (2) My Own History method (MOHi), and (3) Collective History method (CoHi). The first method only considers the website structure and lexical and semantic knowledge of the pages. The second method additionally considers the information related to the pages that the user is visiting in the current session. Finally, the Collective History Method considers the same information as the two previous methods as well as navigational paths (routes) followed by previous visitors of the website. Besides, the performance of the different methods is analyzed under different configurations, which represent different contexts, by means of a wide set of experiments and considering the website of the Comune di Modena in Italy (http://www.comune.modena.it).

The remainder of this paper is structured as follows. Firstly, some related work is studied and analyzed in Sect. 2. Secondly, the different proposals to recommend web pages in large web sites are described and analyzed in Sect. 3. After that, in Sect. 4 the results of a set of experiments to evaluate the performance of the approaches are described. Finally, some conclusions and future work lines are presented in Sect. 5.

2 Related Work

Some works tackle the problem of web page recommendation in a general context, aiming at providing the user with interesting web pages that could fit his/her interests (e.g., [1,2,21,25]). For example, [1,2] propose the use of a multiagent system to search interesting articles in the Web in order to compose a personalized newspaper. In [21,25], the idea is to estimate the suitability of a web page for a user based on its relevance according to the tags provided by similar users to annotate that page. The previous works do not explicitly consider the notion of user session, as their goal is just to recommend web pages to a user independently of his/her current navigation behavior within a specific web site, i.e., the current context of the user.

Other approaches, such as [7,9,12,24], explicitly exploit user sessions and therefore are closer in spirit to our proposals. The SurfLen system [9] suggests interesting web pages to users based on the sets of URLs that are read together by many users and on the similarity between users (users that read a significant number of similar pages). The proposal described in [12] tackles the recommendation problem within a single e-commerce website and proposes an approach to recommend product pages (corresponding to product records in the website database) as well as other web pages (news about the company, product reviews, advises, etc.); although the recommendation is based only on the web page that

the user is currently visiting and not directly on the previous web pages visited by that user, user historical sessions are also exploited to extract information regarding the pages which are visited together (in one session). The approach presented in [7] is based on clustering user sessions and computing a similarity between user sessions in order to recommend three different pages that the user has not visited (a hit is considered if any of the three recommended pages is the next request of the user); the similarity between two user sessions is computed by considering the order of pages, the distance between identical pages, and the time spent on the pages. Another interesting proposal is introduced in [19], where the recommendation system is based on an ad-hoc ontology describing the website and on web usage information. The recommendation model PIGEON (PersonalIzed web paGe rEcommendatiON) [24] exploits collaborative filtering and a topic-aware Markov model to personalize web page recommendations: the recommendations are not just based on the sequence of pages visited but also on the interests of the users and the topics of the web pages. A web page recommendation system is also proposed in [6], but that proposal focuses exclusively on the domain of movies. Movie web pages are clustered by using a weighted k-means clustering algorithm, where pages visited by many users are given higher weights (more importance in the clustering). To recommend new movie web pages to a user, the current active navigation session of the user (web pages that he/she has recently visited) is compared (by using a similarity measure) with the clusters of the movie web pages previously obtained.

There are also some proposals that generalize the problem of web page recommendation to that of web personalization (e.g., see [8,18]). The goal of web personalization is rather to compute a collection of relevant objects of different types to recommend [18], such as URLs, ads, texts, and products, and compose customized web pages. So, a website can be personalized by adapting its content and/or structure: adding new links, highlighting existing links, or creating new pages [8]. Interesting surveys on web mining for web personalization are also presented in [8,13]. However, this kind of approaches require users to be registered in the web site and they need to create and maintain profiles for the different users.

As compared to the previous works, we aim at solving the problem of next URL recommendation within a single web site by exploiting only a limited amount of information available in previous historical user logs. For example, we do not assume that information about the times spent by the users at each URL is available, which may be important to determine the actual interest of a web page (e.g., see [15,22]). Similarly, we do not assume that users can be identified (i.e., they are anonymous), and so it is not possible to extract user profiles. Instead, we propose several methods that require a minimum amount of information and evaluate and compare them in a real context. The methods proposed are also lightweight in the sense that they do not require a heavy (pre-) processing such as semantic extraction from the contents of web pages or the creation and maintenance of indexed structures such as inverted indexes on the content of the web pages.

Finally, a number of other approaches are based the knowledge extracted from URLs. Among them, in [23] the authors applied named entity recognition techniques to URLs with the aim of effectively annotating the contents of the webpages and in [10], websites are clustered on the basis of their URLs.

3 Study of Techniques for Recommending Web Pages

The goal of the recommendation approaches proposed in this paper is to provide the user with a ranked list of suggested URLs (available within a potentially-large website) by considering the context of the user (e.g., the URLs that he/she is currently visiting), structural information about the website, and statistical information available on the logs of the web servers where the site is located. The goal of the application is to recommend pages where the content is similar/related to the content offered by the web page that users are viewing at a specific moment. We assume that users behave rationally and that the exploration performed by them has a purpose (i.e., they are looking for information on a specific topic).

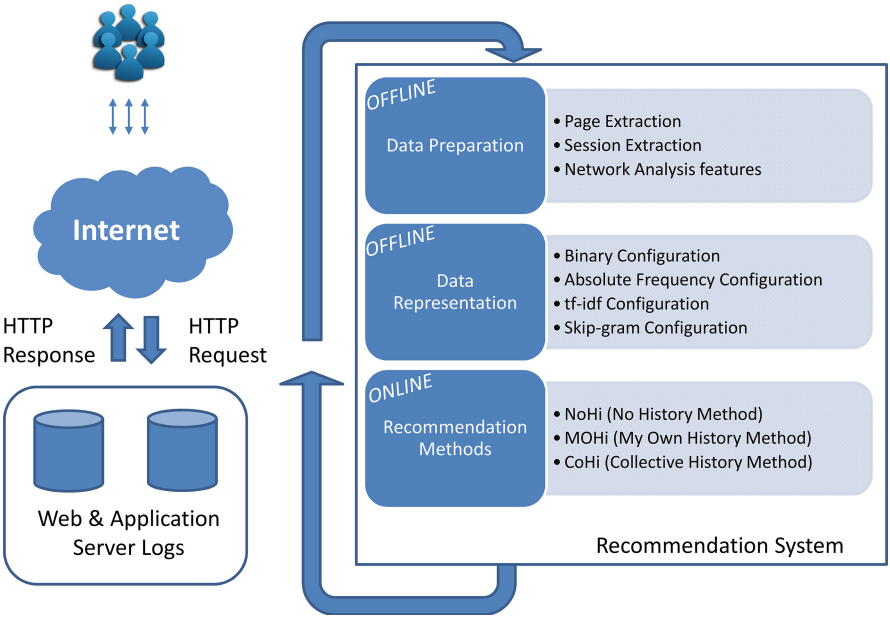

Fig. 1. Functional architecture.

In this section, firstly, models and structures to represent the context of the user, and the content and structure of the website, are presented. After that, three proposed methods (No History Method – NoHi, My Own History Method –

MOHi, and Collective History Method – CoHi) to perform the recommendation are described in detail.

The methods have been implemented in prototypes and evaluated in a real scenario. All the prototypes are built according to the same functional architecture, shown in Fig. 1, where the recommending task is divided into three steps. The first two steps are executed offline and consist in extracting the information about the visited pages in the users' sessions and their representations with proper matrices. We developed and compared four possible techniques (three based on sparse vectors, one on dense vectors) for representing the users' navigation paths in the website as described in Sect. 3.1. In the third step, these different models and structures are used by three methods, as introduced in Sect. 3.2, to generate recommendations.

3.1 Representation of the User Context and the Website

We modeled the users' interactions with the website by adopting and experimenting both sparse and continuous representations of the navigation paths. In particular, by taking as inspiration different classic Information Retrieval (IR) models, we proposed three "bag-of-word" matrix adaptations where the rows represent the different URLs of the website being explored and the columns the vocabulary of those URLs (i.e., all the words that appear in the set of URLs) to model the content and the structure of the website (see Fig. 2). For example, if we consider the URL http://europa.eu/youreurope/citizens/travel/passenger-rights/index_en.htm of the official website of the European Union, then the terms "your europe", "citizens", "travel", "passenger rights", "index" and "en" are part of the vocabulary of the URLs of the website. In this way, the semantic and the lexical content of the web pages is indirectly considered, as it is supposed that the name of the web pages is not random and that the developers follow some kind of convention. Moreover, the website structure is also taken into account, as the categories and nested sections used to organized the website are usually reflected in the paths of the web pages.

	F_1	F_2	F_3	...	F_m
P_1	w_{11}	w_{12}	w_{13}	...	w_{1m}
P_2	w_{21}	w_{22}	w_{23}	...	w_{2m}
P_3	w_{31}	w_{32}	w_{33}	...	w_{3m}
...	...	...	...	...	...
P_n	w_{n1}	w_{n2}	w_{n3}	...	w_{nm}

Fig. 2. Matrix represention of a website.

The user's context is modeled by the vector that represents the web page that he/she is currently visualizing, thus this vector is equal to the row corresponding to the URL of the web page in the matrix representing the website.

To give a value to the different components of the vector representing the user context and the matrix representing the website, classic IR models are considered again as inspiration and the following configurations were analyzed:

- Binary configuration. This configuration is inspired by the Boolean IR model. Thus, each element in the matrix (or vector) indicates whether the URL considered (the row of the matrix or the vector representing the user context) contains (value 1) or does not contain (value 0) the keyword (the term) corresponding to the column of the matrix.
- Absolute-frequency configuration. This configuration is inspired by the first Vector Space IR models. Thus, each element in the matrix (or vector) indicates how many times the keyword corresponding to the column of the matrix appears in the URL considered (the row of the matrix or the vector representing the user context), i.e., the absolute frequency (or raw frequency) of the term in the URL. For example, if we consider the URL http://www.keystone-cost.eu/meeting/spring-wg-meeting-2015/ and the keyword "meeting" then the value of the element corresponding to the column of the term "meeting" is 2. The absolute frequency of a term i in a URL j is represented by $f_{i,\,j}$. So, in this case $f_{meeting,\ www.keystone-cost.eu/meeting/spring-wg-meeting-2015/} = 2$
- TF_IDF configuration. This configuration is inspired by more modern Vector Space IR models, where the length of the documents and the content of the corpus analyzed are considered. Thus, in this case, the length of the URLs and the vocabulary of the set of URLs of the website are considered to define the value of each element of the matrix. In more detail, each element in the matrix (or in each vector) is the product of the relative *Term Frequency* (TF) of the keyword corresponding to the column of the matrix in the URL considered (the row of the matrix) and the corresponding *Inverse Document Frequency* (IDF), i.e., the number of URLs in which that keyword appears. In more detail, $w_{ij} = TF_{ij} * IDF_i$ where

$$TF_{ij} = \frac{f_{i,\,j}}{maximum(f_{i,\,j})} \tag{1}$$

$$IDF_i = \log N/n_i \tag{2}$$

where N is the number of URLs of the website and n_i is the number of URLs where the term i appears.

The previously introduced matrices are both high dimensional (the number of columns is equal to the number of terms existing in the URLs of the website, and the number of rows is equal to the number of web pages available on the website) and sparse (typically, the URL of a web page only contains a limited number of terms of the vocabulary considered as columns), thus only few entries assume values different from zero. An alternative is to use short, dense vectors/matrices (of length 50–2000 columns) that can be efficiently and effectively

computed with machine learning techniques. The development of techniques for using dense matrices in NLP for representing words and documents has recently become popular thanks to works published in [14,16], where a neural network-based model is exploited for building dense and concise word representations in a vector space. Two architectures have been proposed: the CBOW (i.e., continuous bag-of-words) architecture that builds a model that is able to predict the current word based on a context with a parametric dimension, and the Skip-gram model which can support the predictions of surrounding words given the current word. Furthemore, as shown in [17], it was found that similarity of word representations adopting Skip-gram and C-BOW models goes beyond simple syntactic regularities. Using a word offset technique where simple algebraic operations are performed on the word vectors, it was shown for example that vector("King") - vector("Man") + vector("Woman") results in a vector that is the closest to the vector representation of the word Queen. These regularities can be exploited in our scenario for building meaningful and summarized representations of all the pages visited during a session. In this paper, we selected to adopt a Skip-gram model to represent sessions and URLs and our prototypes relies on the gensim[3] library, which provides an implementation of the model in Python. The library requires the model to be trained through "sentences" and generates a dense matrix representing the words used in them. In our scenario, we consider user sessions as sentences and pages browsed in the sessions as words. The result is a model where rows are the URLs of the webpages and columns describing features are exploited for predicting the pages interesting for a user. The Skip-gram model allows also the users to set a number of parameters. We experimented the recommending system with models having different dimensions, trained taken into account contexts with different sizes, and URLs occurring in a predefined number of sessions at least. The results of our experiments are shown in Sect. 4.

3.2 Methods

Three methods proposed to perform web page recommendation in large web sites are described and compared in the following:

– *No History method (NoHi).* In this method, only the current user context is considered, i.e., this method takes into account the information of the web page that the user is currently visualizing to make the recommendation but it does not consider the previous pages that the user has already visited in his/her current session. Thus, the pages recommended to the user are selected by computing the similarity between his/her current state represented by the vector of the URL of the web page visualized and the remaining URLs of the website (the rows of the matrix representing the website in Fig. 2). The most similar URLs to the current user's state are recommended by using as measurement the cosine similarity. According to the literature, this method can be classified as a "content-based recommender system".

[3] https://radimrehurek.com/gensim/.

$$\mathbf{S}_{\mathbf{k}_{\mathbf{history}}} = \mathbf{P_1}\,[w_{11}\ w_{12}\ w_{13}\ \ldots\ w_{1m}] \oplus \mathbf{P_2}\,[w_{21}\ w_{22}\ w_{23}\ \ldots\ w_{2m}] \oplus \ldots \oplus \mathbf{P_k}\,[w_{k1}\ w_{k2}\ w_{k3}\ \ldots\ w_{km}]$$

Fig. 3. User historical context.

– *My Own History method (MOHi)*. In this method, the current web page visited by the user and also the web pages visited in his/her current session (i.e., his/her history) are considered to make the recommendation. Furthermore, the number of pages previously visited taken into account can be limited to a certain number $K_{history}$, which is a configuration parameter of the system. In this case, the user context is modeled as the result of an aggregate function of the already-visited web pages. In this proposal, we adopted two ways for representing the web pages visited in a session: (1) we approximated a session as the sum of the vectors representing its constituent web pages for evaluating recommending systems based on "sparse" representations; and (2) we considered the webpages in each session as a cluster of vectors and the resulting "centroid" as the vector describing the session for evaluating recommending systems based on "dense" vector representations. Nevertheless, any other aggregate function such as a weighted sum (see Fig. 3) could be used. The recommendation is performed in a way similar to the previous method (NoHi). Thus, the aggregated vector is compared with the URLs of the website (the rows of the matrix representing the website in Fig. 2) and the most similar URLs are recommended. This method can also be classified as a "content-based recommender system".
– *Collective History method (CoHi)*. In this method, the history of the user in the current session is also considered. The history is modeled as a list where the items are the different URLs corresponding to the pages that the user has visited. Moreover, this method uses the previous sessions of other users to recommend the web pages. The sessions of the other users are built by extracting information from the logs of the web server of the website and by considering the following rules:

  - A session lasts at most 30 min.
  - A session has to have at least 5 items (i.e., the user has to have visited at least 5 web pages of the website in a session).

  In more detail, matrixes where rows represent the different sessions of the previous users of the website and columns represent the vocabulary of the URLs of the website are built in an analogous way to the previous methods (NoHi method and MOHi method). Aggregated vectors containing all the keywords of the URLs visited during the sessions of the users are built. Those aggregated vectors are built by a simple addition of all the weights of the vectors corresponding to the URLs of the resources visited during the session.

Nevertheless, a weighted sum where for example, the URLs visited initially have less importance than the URLs visited at the end of the session could be applied. After that, the list that models the current session of the user is compared with the sessions of previous users and the top-k similar sessions are retrieved according to the cosine distance. Now, for suggesting the web pages from the top-k sessions we adopt a voting system based on a simple heuristic rule. In particular, we extract all the pages from the sessions and we weigh them according to the position of the session. The rule we follow is that the pages extracted from the top-1 session are weighted k times more than the ones in the k-th session retrieved. The weights in the web pages are then added up, thus generating their rank. Since it exploits the knowledge provided by the navigation of other users, this method can be classified as an "item-based collaborative filtering" recommendation system.

Each method has been implemented to work with the different data configurations described previously. Note that, as shown in Fig. 4, the CoHi method is not applicable to the skip-gram configuration.

	NoHi	MOHi	CoHi
Binary Configuration	X	X	X
Absolute Frequency Configuration	X	X	X
tf-idf Configuration	X	X	X
Skip-gram Configuration	X	X	N.A.

Fig. 4. Approaches developed and compared in the paper.

4 Experimental Evaluation

In this section, we present the experimental evaluation performed to evaluate the proposed methods for web page recommendation. Firstly, in Sect. 4.1, we focus on the dataset used. Then, in Sect. 4.2, we describe the experimental settings. Finally, the results of the experimental evaluation are presented and analyzed in Sect. 4.3.

4.1 Dataset

The "Comune di Modena" Town Hall website[4] has been used in our experiments. It is the official website of an Italian city, having a population of about 200000 citizens. The website visitors are mainly citizens looking for information about

[4] http://www.comune.modena.it.

institutional services (schools, healthcare, labour and free time), local companies that want to know details about local regulations, and tourists interested in information about monuments, cultural events, accommodations and food. To understand the main features of the dataset, we performed two complementary analyses: first, we analyzed the website structure to evaluate the main features of the dataset independently of its actual exploitation by the users; and second, we evaluated the users' behaviors in 2014 by analyzing the logs of the accesses. For achieving the first task, a crawler has been built and adapted for extracting some specific metadata (URL, outgoing links, creation date, etc.) describing the pages. A graph where the web pages are nodes and the links between them are direct edges has been built. The graph representation allowed us to apply a number of simple statistical and network analysis to obtain some details about the website. The results we obtained show that this is a large website composed of more than 13000 pages, classified into more than 30 thematic areas. The average in-degree and out-degree of the pages (i.e., the average number of incoming and outgoing links) is around 13 (the links in the headers and footers have not been computed). This value shows that pages are largely interconnected with each other. Despite the large number of pages, the diameter of the graph is small (8), which means that in the worst case a page can reach another page following a path that crosses 8 other pages. The average path length is 4.57. The modularity of the graph is 0.812. This is a high value that shows that the pages in the website are strongly connected in modules which are poorly coupled with each other. According to this value, the website seems to be well designed with a fixed structure and the information classified in defined sections. Finally, the average clustering coefficient (AAC) is 0.296. This values shows that the graph nodes are not highly connected (a graph completely connected has ACC equal to 1). This value seems to show a countertrend with respect to the other measures that show strong connection levels. As reported by the modularity value, this AAC low degree is normal, since the website is organized in strongly connected modules loosely coupled with each other, and it contains thousands of pages only hierarchically connected.

This analysis was complemented by the analysis of the logs showing the real website usage by the users. In 2014, the number of sessions[5] computed was more than 2.5 millions. The average length of the session is 2.95 pages. Around 10000 pages (72.29% of the overall number of pages) have been visited by at least one visitor in 2014. Only 2809 sessions (0.11% of the overall number of sessions) include in their page the "search engine page" or do not follow the direct links provided in their pages. This demonstrates the quality of the structural design of the website.

4.2 Experimental Settings

In our experiments, we considered the logs from the website limiting our focus on sessions composed of at least 5 pages. The sessions satisfying this requirement

[5] As described in Sect. 3.2, a session includes the pages which are visited by the same user, i.e., the same IP address and User-Agent, in 30 min.

are 303693, 11% of the overall amount. The average length of these sessions is 7.5 pages.

The methods based on sparse matrices have been experimented with a vocabulary of terms built by stemming the words (we adopted the usual Porter stemming algorithm) extracted from the URLs. The vocabulary is composed of 5437 terms representing single words. For improving the accuracy, we added 23555 terms to the previous set, by joining each two consecutive words in the URLs. The methods based on dense vectors have been experimented with a large number of parameter settings, to obtain matrices with different dimensions (50, 100, 250, 500, 1000, and 2000 features), trained taken into account contexts composed of 3 and 6 words, and for words occurring in at least 1 and 10 sessions.

For evaluating the predictions, we divided the pages of a session in two parts: we considered the first two thirds of the web pages in a session as representing the current user's navigation path (we called these pages as the *set of navigation history*), and the remaining one third as the ground truth, i.e., the *set of correct results*. Therefore, our approaches take for each session the set of navigation history as input and provide a recommended page. Only if the page is in set of correct results the result is considered as good.

The following configurations are also considered to decide the types of web pages that can be recommended:

- **No_Exclusion**. This is the general case where URLs that the user has already visited in the current session can also be suggested. Notice that, in this case, the URL where the user is in a specific moment can be also suggested, i.e., the suggestion in this case would be to stay in the same page and not to navigate to another one.
- **Exclusion**. URLs that the user has already visited in the current session cannot be suggested. In this way, the recommendation of staying in the same page is avoided. Moreover, with this configuration, navigating to a previously visited page or the home page of the website is never recommended, despite the fact that coming back to a specific web page already visited during a session is a typical pattern of the behavior of web users.
- **Sub_No_Exclusion**. The difference between this configuration and the one called Exclusion is that we consider only the sessions with no repeated web pages in the set of navigation history. This reduces the number of sessions used in the experiments to 107000. In this configuration, we aim at comparing the performance of our proposal with the one of a typical recommending system. These systems usually do not to recommend items already known/owned by the users. Nevertheless, in the context of websites it is normal that people navigate the same pages multiple times. For this reason in this configuration we consider only cases where in the navigation history there are no pages visited several times in the same sessions. The same constraint is not applied in the set of correct results where we can find pages which are also part of the navigation history (pages already visited).

– **Sub_With_Exclusion**. The difference between this configuration and the one called Sub_No_Exclusion is that here we remove sessions containing repeated web pages independently of their position in the session. In this case, we aim at perfectly simulating the behavior of a typical recommending system.

Note that, for the creation of the matrixes we did not exploited all the logs provided by the web server. Actually, logs have been split into two groups: the first one consists of two thirds of the pages and is used as a training set (i.e., they are used to create the matrices), and the remaining 1/3 of the data is used as a test set (i.e., they provide the sessions used to evaluate the performance of the method). In our experiments, the logs of the 20 first days of each month are considered as training sets while the logs of the last 10 days of each month are considered as test sets.

4.3 Results of the Experiments

Table 1 shows the accuracy of our three approaches based on sparse representations and computed according to the experimental setting defined in the previous section. In particular, Table 1(a) shows the accuracy obtained by the NoHi method, Table 1(b) the accuracy of the MOHi method and finally Table 1(b) the accuracy of CoHi method. Each column of the tables represents one of the configurations introduced in Sect. 3.1 for weighting the matrix that represents the pages visited by the users. In particular, the results applying absolute-frequency, binary, and TF_IDF configurations are shown, in the first, second and third column, respectively.

Table 1. Accuracy achieved in the experiments with sparse representations.

(a) Accuracy on the NoHi method

Configuration	Abs. Freq	Binary	tf_idf
No_Exclusion	0.204	0.21	0.218
Exclusion	0.125	0.130	0.133
Sub_No_Exclusion	0.235	0.243	0.256
Sub_With_Exclusion	0.242	0.252	0.264

(b) Accuracy on the MOHi method

Configuration	Abs. Freq	Binary	tf_idf
No_Exclusion	0.397	0.417	0.467
Exclusion	0.095	0.101	0.101
Sub_No_Exclusion	0.178	0.186	0.194
Sub_With_Exclusion	0.172	0.186	0.188

(c) Accuracy on the CoHi method

Configuration	Abs. Freq	Binary	tf_idf
No_Exclusion	0.584	0.587	0.595
Exclusion	0.192	0.194	0.203
Sub_No_Exclusion	0.310	0.314	0.332
Sub_With_Exclusion	0.360	0.363	0.384

The experiments show that the accuracy of the methods NoHi and MOHi is only partially satisfactory. Moreover, the MOHi approaches suffer from some

Table 2. Accuracy achieved in the experiments with dense representations.

(a) Accuracy on the NoHi method

Num Features	Context	Min Word	Accuracy
50	3	1	0.284
50	6	1	0.293
50	3	10	0.287
50	6	10	0.286
100	3	1	0.285
100	6	1	0.302
100	3	10	0.285
100	6	10	0.287
250	3	1	0.284
250	6	1	0.284
250	3	10	0.294
250	6	10	0.284
500	3	1	0.279
500	6	1	0.283
500	3	10	0.284
500	6	10	0.277
1000	3	1	0.281
1000	6	1	0.286
1000	3	10	0.282
1000	6	10	0.283
2000	3	1	0.281
2000	6	1	0.278
2000	3	10	0.293
2000	6	10	0.281

(b) Accuracy on the MOHi method

Num Features	Context	Min Word	Accuracy
50	3	1	0.387
50	6	1	0.390
50	3	10	0.379
50	6	10	0.389
100	3	1	0.402
100	6	1	0.410
100	3	10	0.398
100	6	10	0.400
250	3	1	0.400
250	6	1	0.407
250	3	10	0.398
250	6	10	0.405
500	3	1	0.404
500	6	1	0.408
500	3	10	0.395
500	6	10	0.404
1000	3	1	0.403
1000	6	1	0.407
1000	3	10	0.394
1000	6	10	0.404
2000	3	1	0.397
2000	6	1	0.406
2000	3	10	0.396
2000	6	10	0.406

noise except in the *No_Exclusion* configuration generated by the user history. Conversely, the accuracy obtained by the application of the CoHi method is good enough for testing the approach in a real environment and is in line with most of the existing recommender systems evaluated in the literature. Moreover, an analysis of the users' sessions show that users typically visit the same pages several times, thus the better results obtained with the No_Exclusion settings. Finally, the experiments in scenarios where both training and testing sessions do not contain repeated visit to the same pages in the same session do not show high accuracy due to the reduced number of instances found in the logs.

Table 2 shows the results obtained evaluating the different settings of the dense representations of URLs. Note that only the outcomes achieved for the *No_Exclusion* configuration (the most general one) and related to the NoHi and CoHi methods (the only techniques applicable to dense matrices) have been reported.

The approaches exploiting dense representations and in particular the ones based on the MOHi methods obtain accuracy values about 40% thus demonstrating effectiveness and performance in line with most of the existing recommender systems. The results show that the accuracy value is not really dependent on

the selected parameters: low dimensional dense representations obtain accuracy results close to the ones obtained with higher dimensional matrices and also high dimensional sparse matrices. This is an interesting result, since it makes the approaches based on dense representation usable more efficiently in real world scenarios, where time performance and reduced computational power can limit the usability of high dimensional vectors.

Finally, we can observe that evaluating a recommender system against logs is unfair. Doing it, we assume that the interesting pages for the users are only the ones that they have really visited. This is not always true, since the suggestions performed by the system can "alter" and "drive" the users' interests. Moreover, some users could have not found the information that they needed. In other words, it would be similar to evaluating a recommender system that suggests products in an e-commerce system based only on the actual purchases made by the users. Other products (web pages in our case) can also be interesting for the users (and potentially suggested by our approaches), even if they did not generate a real purchase (a visit in our case) in the historical data available. Therefore, the results shown in Tables 1 and 2 represent the evaluation in the worst possible scenario.

4.4 Using Statistical and Network Analysis for Improving the Accuracy

All the methods proposed recommends and ranks the interesting web pages on the basis of their context, i.e., the pages (the last one or the complete set) visited in the current session. Statistical analysis on the web server logs and structural analysis provided by network analysis applied to the structure of the website can be applied for improving the accuracy of results.

In this work, we have decided to use additional information in a post-process phase where the pages recommended by the methods are filtered and ranked after. In particular, in this phase we experimented the following measures:

- *Number of visitor per page*: the measure, i.e., the overall amount of times the page has been visited by the users, represents the popularity degree of the page.
- *Page Rank*: this measure indirectly shows the popularity degree of a page at the structural level and estimates the importance of a page on the basis of its neighbors.
- *Betweenness*: the measure computes the importance of a node in a network on the bases of the number of the shortest paths passing through it. This is a measure of the ability of a page to connect other pages, i.e., to be a "central point" in the network.
- *Degree*: the measure shows for each page the number of in-going and out-going links, i.e. the strenght of the connection of a page with the rest of the site. It provides a direct indication on the importance of the page with respect to the structure of the overall site.

Table 3. Accuracy achieved in the experiments with sparse representations and the application of a filter.

(a) Accuracy on the NoHi method and tf-idf configuration

Configuration	Vis.	P.R.	Betw	Dgr.
No_Exclusion	0.261	0.211	0.2	0.2
Exclusion	0.154	0.11	0.107	0.11
Sub_No_Excl	0.303	0.23	0.212	0.212
Sub_With_Excl	0.313	0.236	0.218	0.221

(b) Accuracy on the MOHi method and tf-idf configuration

Configuration	Vis.	P.R.	Betw	Dgr.
No_Exclusion	0.454	0.361	0.36	0.357
Exclusion	0.144	0.1	0.097	0.1
Sub_No_Excl.	0.25	0.159	0.157	0.163
Sub_With_Excl.	0.276	0.186	0.185	0.19

(c) Accuracy on the CoHi method and tf-idf configuration

Configuration	Vis.	P.R.	Betw	Dgr.
No_Exclusion	0.487	0.449	0.456	0.451
Exclusion	0.16	0.137	0.142	0.14
Sub_No_Excl	0.263	0.222	0.227	0.223
Sub_With_Excl	0.386	0.264	0.272	0.27

Table 3 shows the accuracy obtained after the application of the filter on the tf-idf configuration (the one where best results have been achieved as reported in Table 1). The accuracy values do not improve in all the scenarios: typically better results are achieved by ranking the results according to the number of visited pages and the Page Rank. Similar values have been experimented considering approaches based on dense representations of the URLs.

4.5 Further Improvements Using Lexical and Semantic Analysis

One possible direction to improve recommendation systems is to further explore the lexical and semantic information provided by URLs constituents. This approach considers available lexical resources, such as WordNet, that encodes specific and tagged relations between lexical units. Note, however, that the productivity and efficiency of this approach are directly related to the number of units and relations in these resources, as well as encoding options concerning sense specification and delimitation (granularity issues [20]). From the available semantic relations in these resources, the ones serving our purposes are the following:

- Synonymy. Although absolute synonymy is a rare phenomenon in natural languages, the fact is that not always users are completely aware of the actual and/or specialized word that is used in a given context. For example, in the considered website the following terms are synonyms *avviso avvertenza*; *avviso annuncio annunzio comunicazione* (i.e., advertisement).
- Meronymy/holonymy. The relation part-whole is quite relevant in the organization of information/concepts (see [11]). In many cases, meronymy relations provide the conceptual line for the organization of the information, replacing

subtyping relations. For example, in the considered website we found that relationships in *comune giunta* (Municipality - Council); *faculty university*.
- Hyponymy/hyperonymy. Given the fact that hyponymy/hyperonymy relations are the typical hierarchical relations for information organization, and thus reflected on the URL constituents, using subtyping relations for recommendation purposes is likely to return redundant or non-relevant nodes, since hyperonym nodes are expected to be part of the path reflected in the URL. For example the URL http://www.comune.modena.it/tributi/imu-imposta-municipale-propria refers to a local tax (imu) which is part of the section "tax office" (tributi).
- Co-hyponymy. Co-hyponyms are sets of words that share a direct hyperonym and that refer to related concepts and, thus, are relevant for recommendation purposes. For example the terms *scuola publica - scuola privata - scuola serale - accademia* describes different kinds of schools.

These relations can be used differently to test and further improve the different recommendation methods considered. For instance, identifying co-hyponymy relations between pages in MOHi and CoHi methods can be used to refine recommendations and avoid redundant pages; synonymy relations can be used to merge columns in the different matrixes, reducing the second dimension of the matrixes and contributing to increasing the systems speed. This direction assumes shallow processing tasks and can be improved by Part-of-Speech tagging, since lexical resources consider this information and it can be of great use for disambiguation tasks. Besides, the combination of Part-of-Speech info and stemming with the semantic information in these type of resources can also lead to identify and morphologically related words, as in http://comune.modena.it/aree-tematiche/lavoro-e-formazione and *.../lavoro-e-impresa/opportunita-di-lavoro/cercare-lavoro-nel-settore-pubblico/lavorare-per-luniversita/* (URLs referring to the employment sector where "lavoro" means "work" –noun and "lavorare" means "to work" –verb), further exploring the linguistic analysis on URLs.

5 Conclusions and Future Work

In this work, we have introduced two content-based recommendation systems (the NoHi and MOHi methods) to suggest web pages to users in large web sites. These methods base their recommendations on the structure of the URLs of the web site. In particular, they take into account the keywords included in the URLs of the web site. Moreover, we have also presented the CoHi method, that we can consider as a hybrid approach between two types of recommendation systems: content-based recommendation and item-based collaborative filtering. This last approach does not only consider the structure of the URLs, but it also considers information provided by previous users (in particular, the sessions of previous users).

The evaluation of the accuracy of the methods in a real scenario provided by the analysis of the logs of the web servers of the "Comune di Modena" web site shows that the approaches, in particular the last one, achieve a good performance

level. Along this work, we have assumed that if a user visits a page, he/she is interested in the content of that page in the web site. However, it is possible that a user visits a page for other reasons (the pages have been provided by a search engine but they do not satisfy the user information need, the user has clicked on a wrong link, etc.). So, analysis taking into account the amount of time the users spend in the pages will be considered to filter data from the logs used to train and valid the proposed methods.

Acknowledgement. The authors would like to acknowledge networking support by the ICT COST Action IC1302 KEYSTONE - Semantic keyword-based search on structured data sources (www.keystone-cost.eu). We also thank the support of the projects TIN2016-78011-C4-3-R (AEI/FEDER, UE), TIN2013-46238-C4-4-R, and DGA-FSE and the Rete Civica Mo-Net from the Comune di Modena for having provided the data exploited in this research.

References

1. Balabanović, M.: Learning to surf: multiagent systems for adaptive web page recommendation. Ph.D. thesis, Stanford University, May 1998
2. Balabanović, M., Shoham, Y.: Fab: content-based, collaborative recommendation. Commun. ACM **40**(3), 66–72 (1997)
3. Bergamaschi, S., Ferrari, D., Guerra, F., Simonini, G., Velegrakis, Y.: Providing insight into data source topics. J. Data Semant. **5**(4), 211–228 (2016)
4. Bergamaschi, S., Guerra, F., Interlandi, M., Lado, R.T., Velegrakis, Y.: Combining user and database perspective for solving keyword queries over relational databases. Inf. Syst. **55**, 1–19 (2016)
5. Cadegnani, S., Guerra, F., Ilarri, S., Carmen Rodríguez-Hernández, M., Trillo-Lado, R., Velegrakis, Y.: Recommending web pages using item-based collaborative filtering approaches. In: Cardoso, J., Guerra, F., Houben, G.-J., Pinto, A.M., Velegrakis, Y. (eds.) KEYSTONE 2015. LNCS, vol. 9398, pp. 17–29. Springer, Cham (2015). doi:10.1007/978-3-319-27932-9_2
6. Chanda, J., Annappa, B.: An improved web page recommendation system using partitioning and web usage mining. In: International Conference on Intelligent Information Processing, Security and Advanced Communication (IPAC 2015), pp. 80:1–80:6. ACM, New York (2015)
7. Gündüz, S., Özsu, M.T.: A web page prediction model based on click-stream tree representation of user behavior. In: Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2003), pp. 535–540. ACM 2003
8. Eirinaki, M., Vazirgiannis, M.: Web mining for web personalization. ACM Trans. Internet Technol. **3**(1), 1–27 (2003)
9. Fu, X., Budzik, J., Hammond, K.J.: Mining navigation history for recommendation. In: Fifth International Conference on Intelligent User Interfaces (IUI 2000), pp. 106–112. ACM (2000)
10. Hernández, I., Rivero, C.R., Ruiz, D., Corchuelo, R.: A statistical approach to URL-based web page clustering. In: Proceedings of the 21st International Conference on World Wide Web, WWW 2012 Companion, pp. 525–526. ACM, New York (2012)

11. Ittoo, A., Bouma, G., Maruster, L., Wortmann, H.: Extracting meronymy relationships from domain-specific, textual corporate databases. In: Hopfe, C.J., Rezgui, Y., Métais, E., Preece, A., Li, H. (eds.) NLDB 2010. LNCS, vol. 6177, pp. 48–59. Springer, Heidelberg (2010). doi:10.1007/978-3-642-13881-2_5
12. Kazienko, P., Kiewra, M.: Integration of relational databases and web site content for product and page recommendation. In: International Database Engineering and Applications Symposium (IDEAS 2004), pp. 111–116, July 2004
13. Kosala, R., Blockeel, H.: Web mining research: a survey. SIGKDD Explor. **2**(1), 1–15 (2000)
14. Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. CoRR, abs/1405.4053 (2014)
15. Lieberman, H.: Letizia: an agent that assists web browsing. In: 14th International Joint Conference on Artificial Intelligence (IJCAI 1995), vol. 1, pp. 924–929. Morgan Kaufmann (1995)
16. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR, abs/1301.3781 (2013)
17. Mikolov, T., Yih, W., Zweig, G.: Linguistic regularities in continuous space word representations. In: Vanderwende, L., III, H.D., Kirchhoff, K. (eds.) Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA, 9–14 June 2013, pp. 746–751. The Association for Computational Linguistics (2013)
18. Mobasher, B., Cooley, R., Srivastava, J.: Automatic personalization based on web usage mining. Commun. ACM **43**(8), 142–151 (2000)
19. Nguyen, T.T.S., Lu, H., Lu, J.: Web-page recommendation based on web usage and domain knowledge. IEEE Trans. Knowl. Data Eng. **26**(10), 2574–2587 (2014)
20. Nirenburg, S., Raskin, V.: Supply-side and demand-side lexical semantics. In: Viegas, E. (ed.) Breadth and Depth of Semantic Lexicons. Text, Speech and Language Technology, vol. 10, pp. 283–298. Springer, Netherlands (1999)
21. Peng, J., Zeng, D.: Topic-based web page recommendation using tags. In: IEEE International Conference on Intelligence and Security Informatics (ISI 2009), pp. 269–271, June 2009
22. Shahabi, C., Zarkesh, A.M., Adibi, J., Shah, V.: Knowledge discovery from users web-page navigation. In: Seventh International Workshop on Research Issues in Data Engineering (RIDE 1997), pp. 20–29. IEEE Computer Society, April 1997
23. Souza, T., Demidova, E., Risse, T., Holzmann, H., Gossen, G., Szymanski, J.: Semantic URL Analytics to support efficient annotation of large scale web archives. In: Cardoso, J., Guerra, F., Houben, G.-J., Pinto, A.M., Velegrakis, Y. (eds.) KEYSTONE 2015. LNCS, vol. 9398, pp. 153–166. Springer, Cham (2015). doi:10.1007/978-3-319-27932-9_14
24. Yang, Q., Fan, J., Wang, J., Zhou, L.: Personalizing web page recommendation via collaborative filtering and topic-aware Markov model. In: 10th International Conference on Data Mining (ICDM 2010), pp. 1145–1150, December 2010
25. Zeng, D., Li, H.: How useful are tags? — An empirical analysis of collaborative tagging for web page recommendation. In: Yang, C.C., et al. (eds.) ISI 2008. LNCS, vol. 5075, pp. 320–330. Springer, Heidelberg (2008). doi:10.1007/978-3-540-69304-8_32