Exploiting Web Sites Structural and Content Features for Web Pages Clustering

Pasqua Fabiana Lanotte[1(✉)], Fabio Fumarola[2], Donato Malerba[1,3], and Michelangelo Ceci[1,3]

[1] University of Bari Aldo Moro, via Orabona 4, 70125 Bari, Italy
{pasqua.lanotte,donato.malerba,michelangelo.ceci}@uniba.it
[2] Unicredit Research and Development, 20100 Milan, Italy
fabio.fumarola@unicredit.eu
[3] CINI - Consorzio Interuniversitario Nazionale per l'Informatica, Bari, Italy

Abstract. Web page clustering is a focal task in Web Mining to organize the content of websites, understanding their structure and discovering interactions among web pages. It is a tricky task since web pages have multiple dimension based on textual, hyperlink and HTML formatting (i.e. HTML tags and visual) properties. Existing algorithms use this information almost independently, mainly because it is difficult to combine them. This paper makes a contribution on clustering of web pages in a website by taking into account a distributional representation that combines all these features into a single vector space. The approach first crawls the website by using web pages' HTML formatting and *web lists* in order to identify and represent the hyperlink structure by means of an adapted skip-gram model. Then, this hyperlink structure and the textual information are fused into a single vector space representation. The obtained representation is used to cluster websites using simultaneously their hyperlink structure and textual information. Experiments on real websites show that the proposed method improves clustering results.

1 Introduction

Since a web page is characterized by several dimensions (i.e. textual, structural based on HTML tags and visual/structural based on hyperlinks) the existing clustering algorithms differ in their ability of using these representations. In particular, algorithms based on textual representation typically group web pages using words distribution [6,13]. These solutions manage web pages as plain text ignoring all the other information of which a page is enriched and turn to be ineffective in at least two categories of web pages: *(i)* when there is not enough information in the text; *(ii)* when they have different content, but refer to the same semantic class. The former case refers to web pages with poor textual information, such as pages rich of structural data (e.g. from Deep Web Databases), multimedia data, or that have scripts which can be easily found also in other pages (e.g. from a CMS website). The latter case refers to pages having the same semantic type (e.g. web pages related to professors, courses, publications) but

M. Kryszkiewicz et al. (Eds.): ISMIS 2017, LNAI 10352, pp. 446–456, 2017.
DOI: 10.1007/978-3-319-60438-1_44

characterized by a different distribution of terms. On the other side, clustering based on structure typically considers the HTML formatting (i.e. HTML tags and visual information rendered by a web browser) [2,3,7]. Algorithms which use these information, are based on idea that web pages are automatically generated by programs that extract data from a back-end database and embed them into an HTML template. This kind of pages show a common structure and layout, but differ in content. However, because tags are used for content displaying, it happens that most of the web pages in a website have the same structure, even if they refer to distinct semantic types. This negatively affect clusters' quality. The above described solutions exploit within-page information. Other algorithms make use of the graph defined by the hyperlink structure of a set of web pages [18,22]. Hyperlinks can be used to identify collections of web pages semantically related and relationships among these collections. In this area, DeepWalk and Line [18,22] are two embedding-based methods that exploit neural networks to generate a low-dimensional and dense vector representation of graph's nodes. DeepWalk [18] applies the skip-gram method on truncated random walks to encode long-range influences among graph's nodes. Still, this approach is not able to capture the local graph structure (i.e. nodes which can be considered similar because are strongly connected). Line [22] optimizes an objective function that incorporates both direct neighbours and neighbours of neighbours. However, both methods (DeepWalk and Line) ignore node attributes (e.g. textual content).

Most of the discussed works analyze contents, web page structure (i.e. HTML formatting) and hyperlink structure almost independently. Over the last decade, some researchers tried to combine several sources of information together. For example, [1,16] combine content and hyperlink structure for web page clustering, [4,7,19] combine web page and hyperlink structure for clustering purposes. This paper is a contribution in this direction. It combines information about content, web page structure and hyperlink structure of web pages homogeneously. It analyzes web pages' HTML formatting to extract from each page collections of links, called *web lists*, which can be used generate a compact and noise-free representation of the website's graph. Then, the extracted hyperlink structure and content information of web pages are mapped in a single vector space representation which can be used by clustering algorithms. Our approach is based on the idea that two web pages are similar if they have common terms (i.e. *Bag of words hypothesis* [23]) and they share the same reachability properties in the website's graph. In order to consider reachability, the solution we propose is inspired by the concept of *Distributional Hypothesis*, initially defined for words in natural language processing (i.e. "*You shall know a word by the company it keeps*") [9] and recently extended to generic objects [11]. In the context of the Web we can translate that citation in *"You shall know a web page by the paths it keeps"*(i.e. two similar web pages are involved in the same paths).

2 Methodology

The proposed solution implements a four steps strategy: in the first step website crawling is performed. Crawling uses web pages' structure information and exploits web lists in order to mitigate problems coming from noisy links. The output of this phase is the website graph, where each node represents a single page and edges represent hyperlinks. In the second step, we generate a link vector by exploiting Random Walks extracted from the website's graph. In the third phase content vectors are generated. In the last one, a unified representation of pages is generated and clustering is performed on such representation.

2.1 Website Crawling

A Website can be formally described as a direct graph $G = (V, E)$, where V is the set of web pages and E is the set of hyperlinks. In most cases, the homepage h of a website represents the website's entry page and allows the website to be viewed as a rooted directed graph. As claimed in [7] *not all links are equally important to describe the website structure.* In fact, a website is rich of noisy links, which may not be relevant to clustering process, such as hyperlinks used to enforce the web page authority, short-cut hyperlinks, etc.... Besides, the website structure is codified in navigational systems which provide a local view of the website organization. Navigational systems (e.g. menus, navbars, product lists) are implemented as hyperlink collections having same domain name and sharing layout and presentation properties. Our novel solution is based on the usage of web lists. This has a twofold effect: from one side it guarantees that only urls useful to the clustering process are considered; on the other side, it allows the method to implicitly take into account the web page structure which is implicitly codified in the web lists available web pages [7,15,24]. Starting from the homepage h, a crawling algorithm iteratively extracts the collection of the urls having same domain of h and organized in *web lists.* Only web pages included in web lists are further explored. Following [15], a web list is:

Definition 1. *A **Web List** is a collection of two or more web elements having similar HTML structure, visually adjacent and aligned on a rendered web page. The alignment is identified on the basis of the x-axis (vertical list), the y-axis (horizontal list), or in a tiled manner (aligned vertically and horizontally).*

Figure 1 shows, in red boxes, web lists extracted from the homepage of a computer science department which will be used for website crawling. Links in box A will be excluded because their domains are different from the homepage's domain. To identify from a web page the set of web lists we implement HyLien [10]. The output of website crawling step is the sub-graph $G' = (V', E')$, where $V' \subseteq V$ and $E' \subseteq E$ will be used for link and content vectors generation steps.

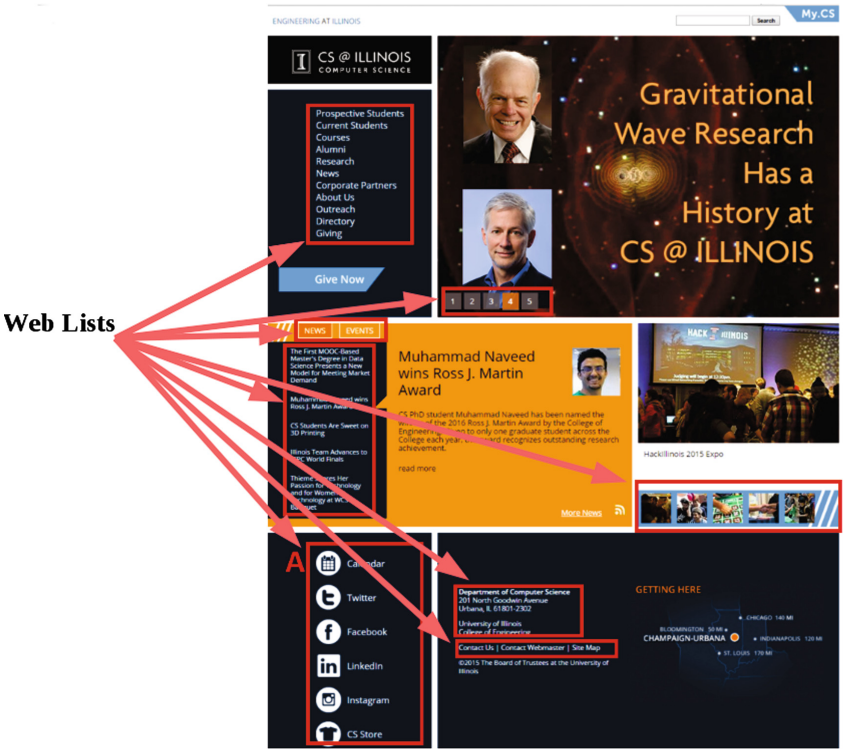

Fig. 1. Web lists extracted from a web page taken from www.cs.illinois.edu (Color figure online)

2.2 Link Vectors Generation Through Random Walks

A random walk over a linked structure is based on the idea that the connections among nodes encode information about their correlations. To codify these correlations we use the Random Walk with Restart (RWR) approach. RWR is a Markov chain describing the sequence of nodes (web pages) visited by a random walker. Starting from a random point i, with probability $(1 - \alpha)$ a walker walks to a new, connected neighbor node or, with probability α, it restarts from i.

Inspired by the field of information retrieval, we model a web page as a word, that is, a *topic indicator* and, each random walk as a document constituting the natural context of words (i.e. topical unity). Thus, we represent a collection of random walks as a document collection where topics intertwine and overlap. This enable the application of a *distributional-based* algorithm to extract new knowledge [21] from the obtained representation. In our case, we apply the skip-gram model [17], a state-of-art algorithm, to extract a vector space representations of web pages that encode the topological structure of the website. In the skip-gram model we are given a word w in a corpus of words V_W (in our case a web page w belonging to random walks) and its context $c \in V_C$ (in our case web pages in

random walks which appear before and after the web page w). We consider the conditional probabilities $p(c|w)$, and given a random walks collection Rws, the goal is to set the parameters θ of $p(c|w;\theta)$ so to maximize the probability:

$$\underset{\theta}{argmax} \prod_{L \in Rws; w \in L} \left[\prod_{c \in C_L(w)} prox_L(w,c) \cdot p(c|w;\theta) \right] \quad (1)$$

where L is a random walk in Rws, w is a web page in L and $C_L(w) = \{w_{i-k}, \ldots, w_{i-1}, w_{i+1}, \ldots, w_{i+k}\}$ is the set of contexts of web page w in the list L. Moreover, $prox_L(w,c)$ represents the proximity between w and $c \in C_L(w)$. This is necessary since the skip-gram model gives more importance to the nearby context words than distant context words. One approach for parameterizing the skip-gram model follows the neural-network language models literature, and models the conditional probability $p(c|w;\theta)$ using soft-max: $p(c|w;\theta) = \frac{e^{v_c \cdot v_w}}{\sum_{c' \in V_C} e^{v_{c'} \cdot v_w}}$, where v_c, $v_{c'}$ and $v_w \in \mathbb{R}^d$ are vector space representations for c, c' and w respectively (d is defined by the user). Therefore, the optimization problem (1) leads to the identification of the web page and context matrices $W = \{v_{w_i} | w_i \in V_W\}$ and $C = \{v_{c_i} | c_i \in V_C\}$. They are dependent each other and we only use W to represent web pages (coherently with what proposed in [17] for words). The computation of $p(c|w;\theta)$ is computationally expensive due the summation $\sum_{c' \in V_C}$ and thus in [17] are presented *hierarchical softmax* and *negative-sampling approach* to make the computation more tractable. Therefore, given in input to skip-gram model a corpus data composed by the collection of random walks, it returns the matrix W which embeds each web page into a dense and low-dimensional space $\mathbb{R}^d$.

2.3 Content Vectors Generation

Here we describe the process for generating a vector representation of web pages using textual information. Differently from traditional documents, web pages are written in HTML and contain additional information, such as tags, hyperlinks and anchor text. To apply on web pages a bag-of-words representation we need a preprocessing step, in which the following operations are performed: HTML tags removal (however, we maintain terms in anchor, title and metadata since they contribute to better organize web pages [8]); unescape escaped characters; eliminate non-alphanumeric characters; eliminate too frequent (>90%) and infrequent (<5%) words. After preprocessing, each web page is converted in a plain textual document and we can apply the traditional *TF-IDF* weighting schema to obtain a content-vector representation. Due the uncontrolled and heterogeneous nature of web page contents, vector representation of web pages based on content is characterized by high-dimensional sparse data. To obtain a dense and low-dimensional space we apply Truncated SVD algorithm, a low-rank matrix approximation based on random sampling [12]. In particular, given the *TF-IDF matrix* of size $|V'| \times n$ and the desired dimensionality of content vectors m, where $m << n$, the algorithm returns a matrix of size $|V'| \times m$.

2.4 Content-Link Coupled Clustering

Once the content vector $v_c \in \mathbb{R}^m$ and the link vector $v_l \in \mathbb{R}^d$ of each web page in V' have been generated, the last step of the algorithm is to concatenate them in a new vector having dimension $m+d$. Before the concatenation step we normalize each vector with its Euclidean norm. In this way we ensure that components of v_l having highest weights are as important as components of v_c having highest weights. The generated matrix preserves both structural and textual information and can be used in traditional clustering algorithms based on vector space model. In this study we consider K-MEANS and H-DBSCAN [5] because they are well known and present several complementary properties.

Table 1. Description of websites

Website	#pages	#edges	#edges using web lists	#clusters
Illinois	563	9415	5330	10
Oxford	3480	44526	35148	19
Stanford	167	12372	30087	10
Princeton	3132	122493	104585	16

3 Experiments

In order to empirically evaluate our approach, we performed validation four computer science department's websites: *Illinois* (cs.illinois.edu), *Princeton* (cs.princeton.edu), *Oxford* (www.cs.ox.ac.ou), and *Stanford* (cs.stanford.edu). The motivation behind this choice is related to our competence in manually labelling pages belonging to this domain. This was necessary in order to create a ground truth for the evaluation of the clustering results. The experimental evaluation is conducted to answer the following questions: (1) which is the real contribution of combining content and hyperlink structure in a single vector space representation with respect to using only either textual content or hyperlink structure? (2) Which is the real contribution of exploiting web pages structure (i.e. HTML formatting) and, specifically, the role of using web lists to reduce noise and improve clustering results? In Table 1 the dimension of each dataset is described. In particular, to correctly analyze the contribution of web lists in the clustering process, we compare only the web pages extracted both by crawling websites using web lists and by traditional crawling (first column of Table 1). Moreover, we report the dimension of the edge set obtained with traditional crawling (second column) and crawling using web lists (third column). Finally the last column describes the number of clusters manually identified by the experts.

We evaluated the effectiveness of the approach using the following measures:

- Homogeneity [20]: each cluster should contain only data points that are members of a single class. This measure is computed by calculating the conditional entropy of the class distribution given the proposed clustering.
- Completeness [20]: all of the data points that are members of a given class should be elements of the same cluster. It is computed by the conditional entropy of the proposed cluster distribution given the real class.
- V-Measure [20]: harmonic mean between homogeneity and completeness.
- Adjusted Mutual Information (AMI): it is a variation of the Mutual Information MI. $MI = \sum_{i \in K} \sum_{j \in C} log \frac{P(i,j)}{P(i)P(j))}$ where C is the set of real classes, K is the set of learned clusters, $P(i, j)$ denotes the probability that a point belongs to both the real class i and the learned cluster j and $P(i)$ is the a priori probability that a point falls into i. MI is generally higher for two clusterings with a larger number of clusters, regardless of whether there is actually more information shared. The Adjusted Mutual Information represents an adjustment of this metric to overcome this limitation.
- Adjusted Random Index (ARI) [14]: it represents a similarity measure between two clusterings by considering all pairs of samples and counting the pairs that are assigned in the same or different clusters in the predicted and true clusterings. $RI = (a + b)/\binom{n}{2}$ where a is number of pairs of points that are in the same class and learned cluster and b is number of pairs of points that belong to different class and learned cluster.
- Silhouette: it measures how similar an object is to its own cluster (cohesion) compared to other clusters (separation). The silhouette ranges from -1 to 1, where a high value indicates that the object is well matched to its own cluster and poorly matched to neighboring clusters.

In order to respond to our research questions we ran our algorithm with different configurations:

- *Text*. We generate a vector space representation, having dimension $m = 120$, using only web pages' textual information;
- *RW-List*. We generate a vector space representation of size $d = 120$ using only hyperlink structure extracted by crawling the website using web lists. We set $\alpha = 1$, $rwrLength = 10$ and $dbLength = 100\,k$;
- *RW-NoList*. We generate a vector space representation of size 120 using only the hyperlink structure obtained with traditional crawling. We ran *rwrGeneration* with the same parameters of RW-List;
- *Comb-Lists*. We combine, as defined in Sect. 2.4, the content vector of size $m = 60$ and hyperlink structure vector of size $d = 60$ generated by crawling the website using web lists.
- *Comb-NoLists*. As in the Comb-Lists, but with a traditional crawler.

Since our goal is not that of comparing clustering algorithms, we set for K-MEANS the parameter K (i.e. total number of clusters to generate) to the number of real clusters, while we set for H-DBSCAN the *minimal cluster size* parameter to 5. Finally, since at the best of our knowledge there is no work which

Table 2. Experimental results

Configuation	Website	Clustering	Homogeneity	Completeness	V-Measure	ARI	AMI	Silhouette
Text	illinois	KMEANS	0.84	0.62	0.71	0.4	0.61	0.33
Text	illinois	H-DBSCAN	0.72	0.53	0.61	0.4	0.5	0.21
RW-Lists	illinois	KMEANS	0.72	0.53	0.61	0.27	0.51	0.42
RW-Lists	illinois	H-DBSCAN	0.81	0.47	0.6	0.18	0.43	**0.43**
RW-NoLists	illinois	KMEANS	0.71	0.52	0.6	0.25	0.5	0.42
RW-NoLists	illinois	H-DBSCAN	0.8	0.45	0.58	0.17	0.41	0.42
Comb-Lists	illinois	KMEANS	**0.9**	**0.69**	**0.78**	**0.54**	**0.68**	0.4
Comb-Lists	illinois	H-DBSCAN	0.83	0.51	0.63	0.27	0.48	0.34
Comb-NoLists	illinois	KMEANS	0.84	0.62	0.71	0.37	0.6	0.38
Comb-NoLists	illinois	H-DBSCAN	0.83	0.52	0.64	0.27	0.49	0.29
Text	Princeton	KMEANS	0.71	**0.59**	**0.64**	**0.68**	**0.58**	0.21
Text	Princeton	H-DBSCAN	0.36	0.31	0.34	0.12	0.28	-0.21
RW-Lists	Princeton	KMEANS	0.56	0.37	0.45	0.27	0.36	0.18
RW-Lists	Princeton	H-DBSCAN	0.49	0.3	0.37	0.12	0.26	-0.05
RW-NoLists	Princeton	KMEANS	0.55	0.36	0.43	0.24	0.35	0.15
RW-NoLists	Princeton	H-DBSCAN	0.48	0.3	0.37	0.1	0.26	-0.09
Comb-Lists	Princeton	KMEANS	0.76	0.54	0.63	0.55	0.53	0.14
Comb-Lists	Princeton	H-DBSCAN	0.47	0.52	0.49	0.36	0.45	**0.37**
Comb-NoLists	Princeton	KMEANS	**0.78**	0.54	**0.64**	0.49	0.53	0.13
Comb-NoLists	Princeton	H-DBSCAN	0.47	0.52	0.49	0.37	0.45	0.38
Text	Oxford	KMEANS	0.74	0.6	0.66	0.48	0.59	0.25
Text	Oxford	H-DBSCAN	0.43	0.41	0.42	0.07	0.37	-0.06
RW-Lists	Oxford	KMEANS	0.65	0.55	0.6	0.48	0.54	0.32
RW-Lists	Oxford	H-DBSCAN	0.6	0.44	0.51	0.26	0.41	0.22
RW-NoLists	Oxford	KMEANS	0.67	0.57	0.62	0.51	0.56	**0.35**
RW-NoLists	Oxford	H-DBSCAN	0.6	0.45	0.51	0.27	0.41	0.18
Comb-Lists	Oxford	KMEANS	0.79	0.67	0.73	**0.56**	0.67	0.34
Comb-Lists	Oxford	H-DBSCAN	0.58	0.49	0.53	0.15	0.47	0.08
Comb-NoLists	Oxford	KMEANS	**0.81**	**0.68**	**0.74**	0.53	**0.68**	0.28
Comb-NoLists	Oxford	H-DBSCAN	0.62	0.53	0.57	0.23	0.51	0.08
Text	Stanford	KMEANS	0.37	0.43	0.39	0.08	0.28	0.3
Text	Stanford	H-DBSCAN	0.18	0.62	0.28	0.07	0.16	0.43
RW-Lists	Stanford	KMEANS	**0.59**	0.58	**0.58**	**0.27**	**0.52**	0.31
RW-Lists	Stanford	H-DBSCAN	0.28	0.4	0.33	0.1	0.22	0.15
RW-NoLists	Stanford	KMEANS	0.47	0.54	0.5	0.14	0.39	0.53
RW-NoLists	Stanford	H-DBSCAN	0.34	0.6	0.43	0.13	0.29	**0.55**
Comb-Lists	Stanford	KMEANS	0.42	0.46	0.44	0.12	0.34	0.22
Comb-Lists	Stanford	H-DBSCAN	0.21	**0.63**	0.31	0.07	0.17	0.46
Comb-NoLists	Stanford	KMEANS	0.53	0.56	0.54	0.17	0.46	0.35
Comb-NoLists	Stanford	H-DBSCAN	0.34	0.51	0.4	0.12	0.28	0.27

uses the skip-gram model to analyze the topological structure of websites, we ran both of skip-gram versions (i.e. hierarchical softmax and SGNS) for generating link vectors. Due to space limitations, we report only results for SGNS (setting the window size to 5), which, in most cases, outperformed hierarchical softmax.

Table 2 presents the main results. In general, the experiments show that best results are obtained combining textual information with hyperlink structure. This is more evident for Illinois and Oxford websites, where content and

Table 3. Wilcoxon pairwise signed rank tests. (+) ((−)) indicates that the second (first) model wins. The results are highlighted in bold if the difference is statistically significant (at p-value = 0.05). The tests have been performed by considering the results obtained with both hierarchical softmax and SGNS skip-gram models.

	Homogeneity	Completeness	V-Measure	Adj Rand index	Adj Mutual info	Silhouette
Text vs Comb	**(+) 0.000**	(-) 0.055	**(+) 0.000**	(+) 0.342	**(+) 0.003**	**(+) 0.020**
RW vs Comb	**(+) 0.002**	**(+) 0.000**	**(+) 0.000**	**(+) 0.000**	**(+) 0.000**	(+) 0.229
NoLists vs Lists	(−) 0.342	(−) 0.970	(−) 0.418	(+) 0.659	(+) 0.358	(−) 0.362

hyperlinks structure codify complementary information for clustering purpose. However, for the Stanford website using the textual information decreases the clustering performance. The importance of combining content and hyperlink structure is confirmed by the Wilcoxon signed Rank test (see Table 3). This behaviour is quite uniform for all the evaluation measures considered.

For the last research question, results do not show a statistical contribution in the use of web lists for clustering purpose (see Table 3). This because analyzed websites are very well structured and poor of noisy links. This can be observed in Table 1, where there is not a valuable difference in terms of edges number between the real web graph and the one extracted using web lists. However, as expected the Completeness is higher for Comb-Lists, confirming that clusters have higher "precision" in the case of crawling based on web lists.

4 Conclusions and Future Works

In this paper, we have presented a new method which combines information about content, web page structure and hyperlink structure in a single vector space representation which can be used by any traditional and best-performing clustering algorithms. Experiments results show that content and hyperlink structure of web pages provide different and complementary information which can improve the efficacy of clustering algorithms. Moreover, experiments do not show statistical differences between results which use web lists and results obtained ignoring web page structure. As feature work we will run our algorithm on different domains and less structured websites in the way to observe whether web lists are really useless in the web page clustering process.

Acknowledgments. We acknowledge the support of the EU Commission through the project MAESTRA - Learning from Massive, Incompletely annotated, and Structured Data (Grant number ICT-2013-612944).

References

1. Angelova, R., Siersdorfer, S.: A neighborhood-based approach for clustering of linked document collections. In: Proceedings of CIKM 2006, pp. 778–779. ACM, New York (2006)
2. Bohunsky, P., Gatterbauer, W.: Visual structure-based web page clustering and retrieval. In: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, pp. 1067–1068. ACM, New York (2010)
3. Buttler, D.: A short survey of document structure similarity algorithms. In: Proceedings of the International Conference on Internet Computing, IC 2004, Las Vegas, Nevada, USA, 21–24 June 2004, vol. 1, pp. 3–9 (2004)
4. Calado, P., Cristo, M., Moura, E., Ziviani, N., Ribeiro-Neto, B., Gonçalves, M.A.: Combining link-based and content-based methods for web document classification. In: Proceedings of the Twelfth International Conference on Information and Knowledge Management, CIKM 2003, pp. 394–401. ACM, New York (2003)
5. Campello, R.J.G.B., Moulavi, D., Sander, J.: Density-based clustering based on hierarchical density estimates. In: Pei, J., Tseng, V.S., Cao, L., Motoda, H., Xu, G. (eds.) PAKDD 2013. LNCS, vol. 7819, pp. 160–172. Springer, Heidelberg (2013). doi:10.1007/978-3-642-37456-2_14
6. Chehreghani, M.H., Abolhassani, H., Chehreghani, M.H.: Improving density-based methods for hierarchical clustering of web pages. Data Knowl. Eng. **67**(1), 30–50 (2008)
7. Crescenzi, V., Merialdo, P., Missier, P.: Clustering web pages based on their structure. Data Knowl. Eng. **54**(3), 279–299 (2005)
8. Fathi, M., Adly, N., Nagi, M.: Web documents classification using text, anchor, title and metadata information. In: Proceedings of the International Conference on Computer Science, Software Engineering, Information Technology, e-Business and Applications, pp. 1–8 (2004)
9. Firth, J.: A synopsis of linguistic theory 1930-55. In: Palmer, F.R. (ed.) Selected Papers of J.R. Firth 1952-59, pp. 168–205. Longmans, London (1968)
10. Fumarola, F., Weninger, T., Barber, R., Malerba, D., Han, J.: Hylien: a hybrid approach to general list extraction on the web. In: Proceedings of the 20th International Conference on World Wide Web, WWW 2011, Hyderabad, India, 28 March - 1 April 2011 (Companion Volume), pp. 35–36 (2011)
11. Gornerup, O., Gillblad, D., Vasiloudis, T.: Knowing an object by the company it keeps: a domain-agnostic scheme for similarity discovery. In: Proceedings of the 2015 IEEE International Conference on Data Mining (ICDM), ICDM 2015, pp. 121–130. IEEE Computer Society, Washington, DC (2015)
12. Halko, N., Martinsson, P.G., Tropp, J.A.: Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions. SIAM Rev. **53**(2), 217–288 (2011)
13. Haveliwala, T.H., Gionis, A., Klein, D., Indyk, P.: Evaluating strategies for similarity search on the web. In: Proceedings of WWW 2002, pp. 432–442. ACM, New York (2002)
14. Hubert, L., Arabie, P.: Comparing partitions. J. Classif. **2**(1), 193–218 (1985)
15. Lanotte, P.F., Fumarola, F., Ceci, M., Scarpino, A., Torelli, M.D., Malerba, D.: Automatic extraction of logical web lists. In: Andreasen, T., Christiansen, H., Cubero, J.-C., Raś, Z.W. (eds.) ISMIS 2014. LNCS, vol. 8502, pp. 365–374. Springer, Cham (2014). doi:10.1007/978-3-319-08326-1_37

16. Lin, C.X., Yu, Y., Han, J., Liu, B.: Hierarchical web-page clustering via in-page and cross-page link structures. In: Zaki, M.J., Yu, J.X., Ravindran, B., Pudi, V. (eds.) PAKDD 2010. LNCS (LNAI), vol. 6119, pp. 222–229. Springer, Heidelberg (2010). doi:10.1007/978-3-642-13672-6_22
17. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. Adv. Neural Inf. Process. Syst. **26**, 3111–3119 (2013)
18. Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: online learning of social representations. In: ACM SIGKDD 2014, KDD 2014, pp. 701–710. ACM, New York (2014)
19. Qi, X., Davison, B.D.: Knowing a web page by the company it keeps. In: Proceedings of the 15th ACM International Conference on Information and Knowledge Management, CIKM 2006, pp. 228–237. ACM, New York (2006)
20. Rosenberg, A., Hirschberg, J.: V-measure: a conditional entropy-based external cluster evaluation measure. In: EMNLP-CoNLL 2007, pp. 410–420 (2007)
21. Sahlgren, M.: The distributional hypothesis. Ital. J. Linguist. **20**(1), 33–54 (2008)
22. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: large-scale information network embedding. In: Proceedings of the 24th International Conference on World Wide Web, WWW 2015, New York, NY, USA, pp. 1067–1077 (2015)
23. Turney, P.D., Pantel, P.: From frequency to meaning: vector space models of semantics. J. Artif. Int. Res. **37**(1), 141–188 (2010)
24. Weninger, T., Johnston, T.J., Han, J.: The parallel path framework for entity discovery on the web. ACM Trans. Web **7**(3), 16:1–16:29 (2013)