The Power of AI in IoT : Cognitive IoT-based Scheme for Web Spam Detection

Aaisha Makkar *Student Member, IEEE,*
Computer Science & Engineering Department
Chandigarh University (Chandigarh), India
Email: aaisha.e8847@cumail.in

Neeraj Kumar *Senior Member, IEEE*
Computer Science & Engineering Department
Thapar Institute of Engineering & Technology
Patiala (Punjab), India
Email: neeraj.kumar@thapar.edu

Mohsen Guizani *Fellow, IEEE*
University of Idaho Moscow
Email: mguizani@ieee.org

***Abstract*—In the modern era, Internet of Things(IoT) plays an important role in connecting the people across the globe. The IoT objects enable the communication and data exchange among each other irrespective of their geographical locations. In such an environment, the Web of Things (WoT) provides the Internet service to the IoT objects. The Internet is mostly accessed by the search engines. The success of search engine depends upon the ranking algorithm. Although, Google is preferred by the maximum Internet users, but still the Google's ranking algorithm, *PageRank* experiences the occurrence of spam web pages. In this paper, the webpage filtering algorithm is proposed which automatically detects the spam web pages. The spam webpages are detected before these are processed by the ranking module of search engines. The machine learning model, i.e., decision tree is used for the validation of the proposed scheme. The ten fold cross validation approach is used to improve the accuracy of model, i.e., 98.2%. The results obtained demonstrate that the proposed scheme has the power of preventing the spam web pages in Cognitive Internet of Things (CIoT) environment.**

***Keywords*—web spam; IoT; AI; CIoT; WoT.**

I. Introduction

In today's era, computing has been changed from hardware to software, touching to sensing and intelligence to smartness. These changes are due to the rapid growth of technologies (IoT, Smart Systems, Protocols etc.). The Internet is the backbone of all these advancements. However, the fourth phase of Internet, i.e., Internet of things (IoT) has connected the world with the help of smart devices, known as *objects*. These objects are intelligent to perform various actions like computing, sensing and communicating. The fifth phase of Internet is the Cognitive Internet of things (CIoT), in which Artificial Intelligence is embedded in these objects. The three dimensional network architecture of CIoT is shown in Fig. 1. This architecture consists of three different planes, i.e., Protocol Plane (PP), Adjusting plane (AP), and Cognitive plane (CP). Protocol plane consists of four different layers as- Intelligent Service Layer (ISL), Information Fusion Layer (IFL), Network Interconnection Layer (NIL), and Information Perception Layer (IPL). All the phases of Internet are summarized in Fig. 2. The successful CIoT device should have the cognitive ability to act intelligently in the environment where objects are deployed. Also, the Internet is mostly accessed using the search engine. Search engine suffers in having accurate search engine result pages (SERPs) due to the presence of spam webpages in the web repository. It is the responsibility of ranking module to provide rank score to the webpages. Thus, giving cognitive power to search engine, can automatically detect spam webpages.

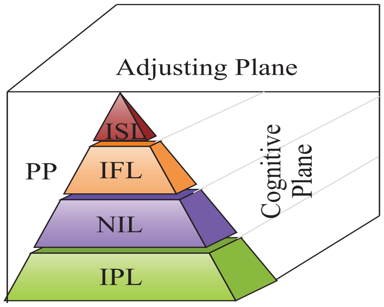

Fig. 1. Cognitive architecture for IoT environment [1].

Ranking module of search engine, ranks the web pages according to their importance for fast search engine result pages (SERP). Google uses PageRank algorithm as the ranking methodology. PageRank is introduced by Brin and Page in 1999 [2]. Due to the presence of large number of websites, the maximum visit count is a challenging task for website developer. The visit count of a website depends upon its rank in SERP. Following are the issues considered by the search engine's ranking module:

1) The PageRank issues are resolved by selecting the appropriate edges of graphs in the polynomial time. This has been successfully demonstrated by imple-

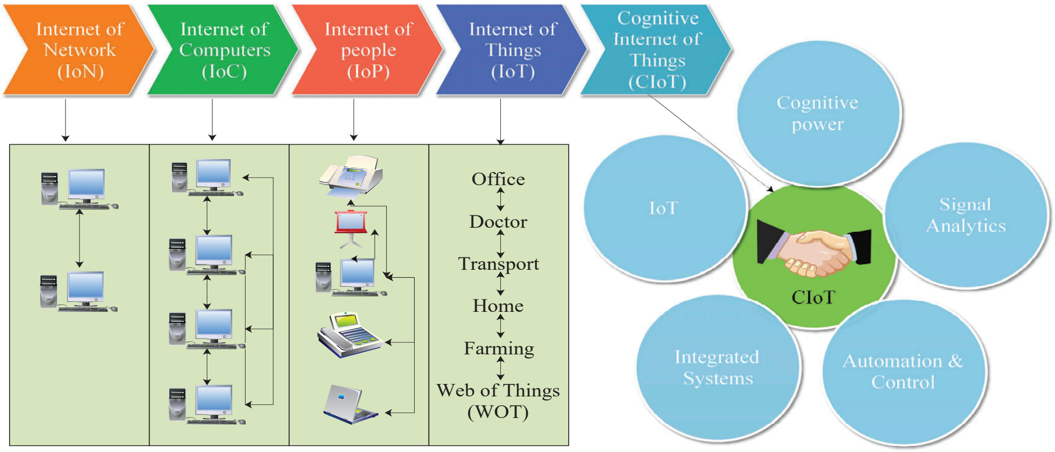

Fig. 2. CIOT with the Internet phases.

menting the Markov chain process as a stochastic shortest path problem. The same selection process is performed with the greedy solution and is called as the PageRank iteration algorithm [3].

2) To decrease the number of iterations of the power method, the web graph can be considered as a tree, which is one solution. The tree is formed by core-tree decomposition followed by applying the GMRES method to a small-tree-width graph [4].
3) *VertexRank* is another algorithm [5] used for assigning weights to the vertices in the graph. In this algorithm, the central vertex with the highest weight is placed first. All of the other vertices are then assigned depending on the depth of graph and number of edges connected. Thus, each vertex holds its position and is placed according to its coefficient. Finally, the position of each vertex is determined via applications of the classical force-directed algorithm [5].
4) An algorithm[6] in which each webpage, with the help of links, can update its own PageRank has been developed. It is known as the randomized algorithm because it works randomly on the data set. The main emphasis of this algorithm is to improve the convergence rate of PageRank values. One agent/webpage updates its value by taking the average of all the values received by it at a particular instant. A webpage can send its score to various out-links and can receive the same from in-links. To reduce the communication and computational load, instead of waiting for the PageRank from all of the links, one page can wait until the convergence of the PageRank and then simultaneously update. This concept is modified by locally updating the PageRank by communicating with the neighbors [6].
5) To reduce the computational load, the web aggregation approach was developed[7]. In this approach, the Web is divided into various groups according to the criteria of links, and each individual group then computes its PageRank. Thus, the computed value is the total value of the whole group, which is distributed among all the group members. In this approach, the request for the PageRank is forwarded only to the outgoing links [7].
6) The improved Monte Carlo (MC) method states that whenever the crawler fetches new data from the Web, it should be implemented in parallel. This leads to the resolution of the PageRank update. MC algorithms can start up either randomly or cyclically, but random start is not as efficient as cyclic start. The information about all the visited pages is maintained by this algorithm [8].
7) Two sampling methods proposed for this purpose are direct sampling in power iteration (DSPI) and adaptive sampling in power iteration (ASPI). Both are based on low-rank approximations of the transition matrix. DSPI performs the sampling of the transition matrix only once and then uses it in the computation process, whereas ASPI performs the sampling of the transition matrix many times, with the adaptive sampling rate being adjusted in the computation process [9].
8) Many algorithms have been introduced to increase the convergence rate of the power method irrespective of the damping factor. The improved PageRank method referred to as the multiplicative splitting iteration method is introduced by using the linear system methodology and multi-splitting iterations [10].
9) The same work has been extended by re-partitioning the workload by processing the values on different processors. The task is accomplished via the implementation of 1D and 2D compression schemes. The 2D scheme performed better in terms of load

partitioning [11].

10) The approach of distributing the web graph into domains of quasi-equivalent vertices and then computing the probability distribution of a random walk was developed. This method computes the PageRank scores with the formulation of intra-host and inter-host steps. It results in a better approximation of PageRank scores [12].

Search engine's ranking issues are still a open research topic, and researchers keep on trying to resolve these. One of the ranking issues, i.e., web page filtering, is addressed in this proposal.

A. Motivation

The search engine result pages (SERP) depend upon the ranking methodology of the search engine. Ranking algorithms are always in a way that it can ignore the spam pages. However, still the spammers try to alter the ranking algorithm to invade the spam pages into the top of SERPs. Many authors have tried to detect the spam pages [13]. There is the need of intelligent system, which can automatically detect spam pages. Such a task has been accomplished in this paper.

B. Contributions

This Section describes the contributions made for the formulation of the proposed scheme.

1) The redundant features are extracted by computing the correlation among features.
2) The dimension space of the dataset is reduced using PCA technique.
3) Validation of the proposed scheme with the support of machine learning model is performed.

C. Organization

The rest of the paper is organized as follows. Section II presents the System model. The proposed scheme is illustrated in Section III. Section IV presents the results and discussion. Finally, the paper is concluded in Section V.

II. System Model

IoT has enabled the exchange of information among devices with the advancement in web. The reliability and efficiency in information extraction depend upon the efficiency of search engine. The websites maintained by search engines includes various phases, such as storage, indexing, and ranking. The cost incurred by the ranking module for ranking the websites, signifies the effectiveness of search engine's methodology. This proposal aims to reduce this computational cost by websites filtering as illustrated below.

$$A_{t,s} = C_t + C_s \tag{1}$$

$$\frac{(\partial)A_{t,s}}{(\partial)t} = \frac{\partial C_t}{(\partial)t} + \frac{\partial C_s}{(\partial)t} \tag{2}$$

$$\frac{(\partial)A_{t,s}}{(\partial)s} = \frac{\partial C_t}{(\partial)s} + \frac{\partial C_s}{(\partial)s} \tag{3}$$

C_t refers to time complexity, C_s refers to space complexity, and $A_{t,s}$ is the sum of time and space complexity. The costs include the computational cost for accessing information from the storage, which involves time and cost complexity. In Eq. 1, $A_{t,s}$ is to be minimized, which is computed by the summation of situational parameters, i.e., time and space complexity. Eq. 2 will minimize A with respect to time and Eq. 3 will minimize A with respect to space.

III. Proposed Scheme

This cost effective model for ranking the websites is built by giving the cognitive power to the objects. This power filters the websites by the detection of spam websites before the ranking module. Webpage filtering has been given attention in literature. Different methods used for webpage filtering are discussed below:

- Grilheres *et al.*, 2004 [14] combined several classifiers for obtaining filtered webpages. The targeted audience for the concerned project is the safety of students by ignoring harmful webpages. The experiments considered the webpages of two languages, i.e., English and French.
- Price *et al.*, 1999 [15] launched a computer program for webpage filtering. The program considers the quality of health information for refining the webpages. The undesirable pages were eliminated with the help of rank score computation by the proposed program. Set of 48 webpages consisting of 9 different medical topics was considered for experiments [15].
- Christopher Lueg, 1998 [16], proposed a recommender system for sharing information among the audience of particular domain. This system recommends the URL of a particular webpage to another user. The system was named as Collaborative Recommend-er Agent (CORA). The same system has been used for the communication within the organization.
- Elhadi *et al.*, 2009 [17], developed an algorithm with Longest Common Sub-sequence (LCS) concept. This algorithm filters the webpages by eliminating the duplicate content. It helped in revising the rank of search engine results.

Analyzing the webpage features can help in detection of spam webpage. Machine learning classifier is the technique used in this paper to find the spam webpages. The webpage

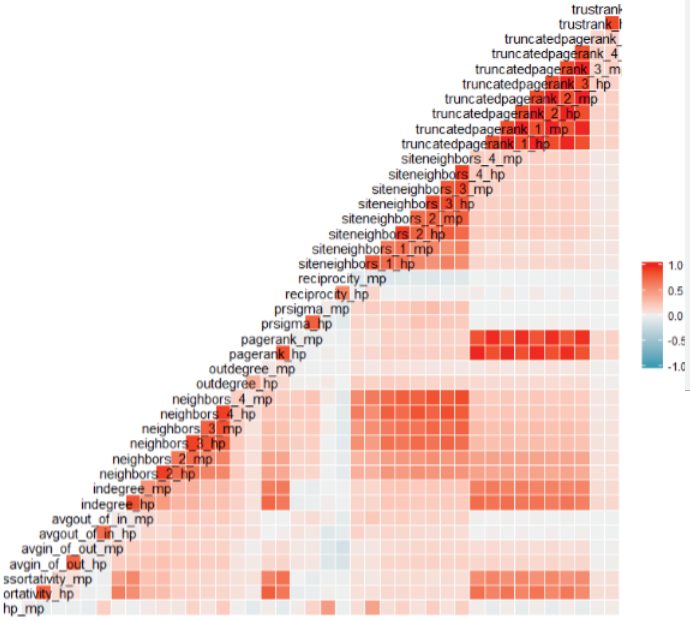

Fig. 3. Correlation among features.

Algorithm 1 Web filtering algorithm

Input:Web matrix
Output: Filtered web pages

```
1:  procedure FUNCTION(Filter)
2:      for $j = 1 \; to \; T$ do
3:          if I and O == 0 then
4:              Discard webpage
5:          end if
6:      end for
7:      for $i = 1 \; to \; T$ do
8:          Calculate correlation(cor) foreach i
9:          if cor_i=-1 then
10:             ignore i                                  ▷ Negatively correlated
11:         elsecor_i=1
12:             consider i                                ▷ Positively correlated
13:         end if
14:     end for
15:     for $k = 1 \; to \; T$ do
16:         $A[m,n] \leftarrow G$                        ▷ Graph containing filtered features.
17:         create $F_{ij} = \begin{cases} 1 \; if(i,j) \in E \\ 0 \; otherwise \end{cases}$
18:     end for
19: end procedure
```

filtering is done after the detection of spam webpages. The whole process of web filtering is done before assigning the rank scores to the webpages by the ranking module. Algorithm for the procedure of web filtering is proposed. Thus, this algorithm reduces the overload of ranking process and leads to whole refining of search engine results.

The approach adopted in Algorithm 1 filtered the web pages by excluding the redundant features. These features were computed by calculating the correlation among features. In steps 2 to 5, the webpages having no in-link and out-link are detected. Such webpages are known as Dangling webpages. In step 8, the correlation among the features is computed, as shown in Fig. 3. If the correlation is -1, it states that the features are perfectly correlated but negatively, i.e., if the value of one feature increases, then

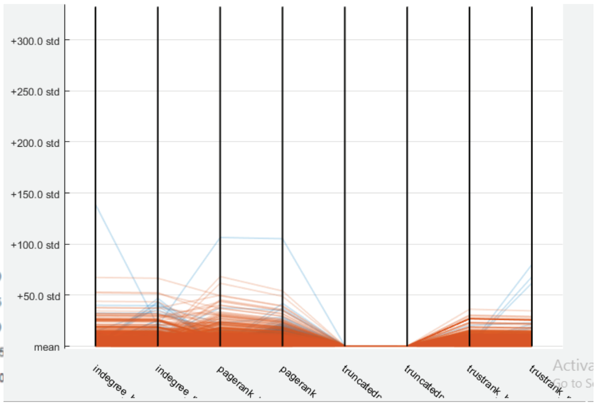

Fig. 4. Parallel coordinates plot.

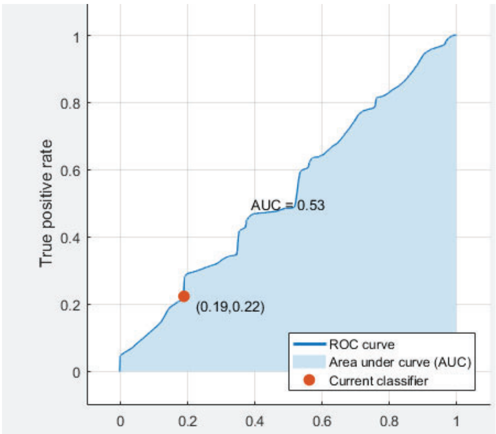

Fig. 5. ROC Curve before applying proposed algorithm (Decision Tree).

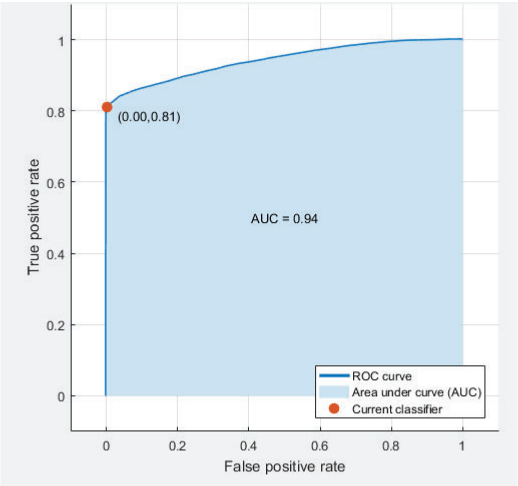

Fig. 6. ROC Curve (Decision Tree).

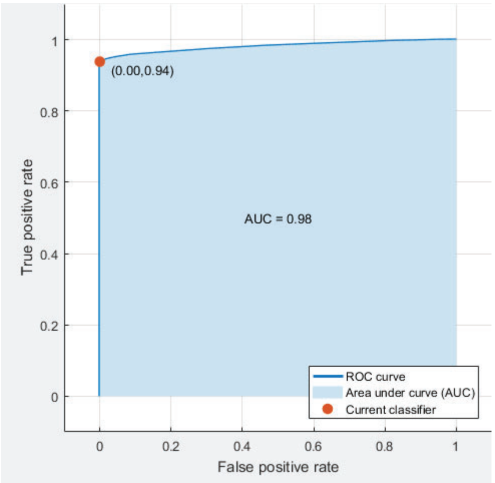

Fig. 7. ROC Curve after applying PCA (Decision Tree).

the value of another feature decreases. If the correlation is 1, it states that the features are perfectly correlated but positively, i.e., if the value of one feature increases, then the value of another feature decreases. Finally, the filtered webpages formed a web graph G as in step 16. Now, the matrix F is created which contains the webpages to be fed into the spam detection classifier. This classifier detects the spam pages.

IV. Results and Discussion

The execution of computing the rank score depends upon the web matrix. The proposed scheme reduces the size of web matrix by filtering the webpages. The reduced size of web matrix helps the ranking module in saving the time and refining the SERPs. The filtered web pages help in the detection of spam web pages.

A. Data Collection

The proposed scheme of webpage filtering and web spam detection has been experimented using the dataset available publicly, i.e., WEBSPAM-UK2007. This data collection is launched by the *Universita degli Studi di Milano*, the Laboratory of Web Algorithms [18]. It is the collection of 110532 hosts with 41 features. The link features are considered for experiments. The Algorithm presented in Section 3 filters the webpages and decreases the size of web matrix using this dataset.

B. Spam Detection Classifier

The data used for the validation of proposed scheme should not be noisy, i.e., the data should be clean enough to be fit in the classifier. So, the dataset is pre-processed before experiments are conducted. The coordinates of few features (PageRank, in-degree, truncated pagerank, trustrank) which are parallel to each other are presented in Fig. 4. The procedure used to refine the data is as follows:

- PCA: Principal Component Analysis (PCA) is used to reduce the dimension of the data. It is the feature extraction process which converts the correlated variables into uncorrelated variables to extract the maximum variance. In other words, it compresses the data without losing its importance. The method *pca()* is invoked from FactomineR package. PCA not only reduces the complexity of data, but also helps in processing the data.
- Feature selection: The features are selected by analyzing after computing the correlation among them. The *ggcorr()* function is invoked from the package GGally package. Correlation between 41 features can not be presented here, so the correlation between two features, i.e., PageRank and indegree is presented in Fig. 8.
- Machine learning model: The filtered data after the feature extraction and feature selection is fed into the algorithm of complex tree (A decision tree with multiple leaves to handle large volume of data). The *fitctree()* function is used by the decision tree. The tree has been trained with three variations as discussed below:
    1) Before webpage filtering: Before pre-processing the data, the original data is fed into the machine learning model (Decision Tree). The prediction speed is 610000 obs/sec. The accuracy obtained is 63.9% as shown in Fig. 5. The training time taken by the system is 39.273 secs.
    2) After webpage filtering, but without applying PCA: The data without being extracted, is fed into the machine learning model (Decision Tree). The prediction speed is 510000 obs/sec. The accuracy obtained is 94.3% as shown in Fig. 6. The training time taken by the system is 33.937 secs.
    3) After webpage filtering and applied PCA: The data after compressing with PCA, is fed into the machine learning model (Decision Tree). The prediction speed is 380000 obs/sec. The accuracy obtained is 98.2% as shown in Fig. 7. The training time taken by the system is 11.339 secs.

It is observed from the experiments that the proposed scheme of web filtering maximizes accuracy in minimum time.

V. Conclusion

To improve the results of search engine and performance of IoT objects, a webpage filtering scheme based on cognitive Internet of Thing is proposed in this article. The proposed scheme automatically detects the spam webpages. The webpage filtering has been done by analyzing the features. The linkage information of the webpages are considered for the experiments. The objective of this proposal is to reduce the computational cost of the ranking module.

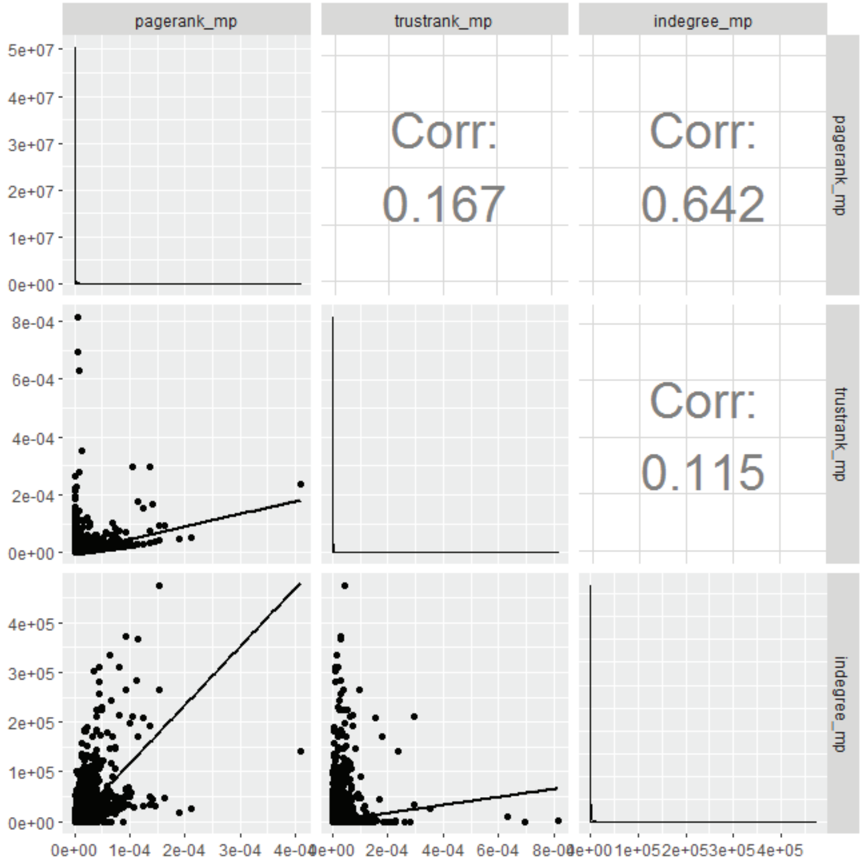

Fig. 8. Correlaton between PageRank and Indegree.

The experimental data has been pre-processed to reduce the complexity. A machine learning model is built for validating the proposed scheme. In future, we would like to explore spam detection techniques for resolving search engine's ranking issues.

References

[1] M. Zhang, H. Zhao, R. Zheng, Q. Wu, and W. Wei, "Cognitive internet of things: concepts and application example," *International journal of computer science issues*, vol. 9, no. 6, pp. 151–158, 2012.

[2] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking: bringing order to the web," 1999.

[3] B. C. Csáji, R. M. Jungers, and V. D. Blondel, "Pagerank optimization by edge selection," *Discrete Applied Mathematics*, vol. 169, pp. 73–87, 2014.

[4] T. Maehara, T. Akiba, Y. Iwata, and K.-i. Kawarabayashi, "Computing personalized pagerank quickly by exploiting graph structures," *Proceedings of the VLDB Endowment*, vol. 7, no. 12, pp. 1023–1034, 2014.

[5] W. Dong, F. Wang, Y. Huang, G. Xu, Z. Guo, X. Fu, and K. Fu, "An advanced pre-positioning method for the force-directed graph visualization based on pagerank algorithm," *Computers & Graphics*, vol. 47, pp. 24–33, 2015.

[6] H. Ishii and R. Tempo, "Distributed randomized algorithms for the pagerank computation," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 1987–2002, 2010.

[7] H. Ishii, R. Tempo, and E.-W. Bai, "A web aggregation approach for distributed randomized pagerank algorithms," *IEEE Transactions on automatic control*, vol. 57, no. 11, pp. 2703–2717, 2012.

[8] K. Avrachenkov, N. Litvak, D. Nemirovsky, and N. Osipova, "Monte carlo methods in pagerank computation: When one iteration is sufficient," *SIAM Journal on Numerical Analysis*, vol. 45, no. 2, pp. 890–904, 2007.

[9] W. Liu, G. Li, and J. Cheng, "Fast pagerank approximation by adaptive sampling," *Knowledge and Information Systems*, vol. 42, no. 1, pp. 127–146, 2015.

[10] C. Gu and L. Wang, "On the multi-splitting iteration method for computing pagerank," *Journal of Applied Mathematics and Computing*, vol. 42, no. 1-2, pp. 479–490, 2013.

[11] A. Cevahir, C. Aykanat, A. Turk, and B. B. Cambazoglu, "Site-based partitioning and repartitioning techniques for parallel pagerank computation," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 5, pp. 786–802, 2011.

[12] A. Z. Broder, R. Lempel, F. Maghoul, and J. Pedersen, "Efficient pagerank approximation via graph aggregation," *Information Retrieval*, vol. 9, no. 2, pp. 123–138, 2006.

[13] A. Makkar and N. Kumar, "User behavior analysis-based smart energy management for webpage ranking: Learning automata-based solution," *Sustainable Computing: Informatics and Systems*, 2018.

[14] B. Grilheres, S. Brunessaux, and P. Leray, "Combining classifiers for harmful document filtering," in *Coupling approaches, coupling media and coupling languages for information retrieval*, 2004, pp. 173–185.

[15] S. L. Price and W. R. Hersh, "Filtering web pages for quality indicators: an empirical approach to finding high quality consumer health information on the world wide web," in *Proceedings of the AMIA Symposium*, 1999, pp. 911.

[16] C. Lueg, "Considering collaborative filtering as groupware: Experiences and lessons learned," in *PAKM*, vol. 98, pp. 16, 1998.

[17] M. Elhadi and A. Al-Tobi, "Duplicate detection in documents and webpages using improved longest common subsequence and documents syntactical structures," in *2009 Fourth International Conference on Computer Sciences and Convergence Information Technology*, 2009, pp. 679–684.

[18] L. B. P. B. M. S. Carlos Castillo, Debora Donato and S. Vigna, "Web spam collections," 2007.