EAI Endorsed Transactions
on Energy Web

Research Article EAI.EU

Semantic Web and Web Page Clustering Algorithms: A Landscape View

Ahmed J. Obaid[1,*], Tanusree Chatterjee[2] and Abhishek Bhattacharya[3]

[1]Faculty of Computer Science and Mathematics, University of Kufa, Iraq.
[2]Techno International NewTown, India.
[3]Institute of Engineering and Management, India.

Abstract

The major evolution of the semantic web has become exchanging data between applications in all domains of activities. Based on this vision, different applications in recent days, e.g. in the fields of community web portals, social networking, e-learning, multimedia retrieval, etc. have been designed. Due to growing number of web services, clustering of web resources becomes a valuable tool for semantic web mining. Clustering of internet objects like Internet web pages' intimate new methods for grouping correlated content for better understanding and satisfies massive user query results in web pages' search. Hence, web pages clustering algorithms should be able to handle massive irregular content and discover knowledge regardless of the web page complexity. These algorithms vary depending on the characteristics and data types. So, choosing the most appropriate algorithm is not an easy process as it should be accurate in terms of time and space complexity. Therefore, this paper rigorously surveys the most important algorithms of different types used for web page clustering. In addition, a comparative analysis of all such algorithms are provided in terms of several parameters. Finally, a brief discussion is provided on why web page clustering is important in emerging era of Semantic Web of Thing (SWoT) applications.

Keywords: Semantic web, web page clustering, cluster analysis, clustering algorithms.

Received on 23 August 2020, accepted on 22 October 2020, published on 18 November 2020

doi: 10.4108/eai.18-11-2020.167099

*Corresponding author. Email: [1]ahmedj.aljanaby@uokufa.edu.iq.

1. Introduction

Today petabytes of data are available in web which are unstructured, structured, and semi structured. Tremendous growth of websites and web contents in the form of text, multimedia messages on the WWW has led to demand of a strategy which can provide knowledge from the vast data scattered over different servers. Accessing and retrieving information from structured data is easy but due to massive growth of web data in past few years, most of the data these content on the WWW is mostly unstructured and human understandable. Hence, the requirements to improve the users searching results from the scattered and massive number of internet web pages is key challenge in existing search engines, which typically aims to sequence the relevant result in a sequential form. Here, semantic web mining comes into action which combines web mining and semantic web towards making the data structured and machine readable thus supporting easier data discovery, data integration, navigation, and automation of tasks. Mining of web content is like techniques of data mining where the applications aim to extract the hidden patterns or features from web pages. The clustering techniques of

semantic web provide query relevant knowledge with feature extraction techniques on the massive and linked data present in WWW [4][5]. The objective of the paper is to introduce the core idea of the commonly used clustering algorithms and analyse the advantages and disadvantages of each one.

Web mining [1] [2] is defined as mining of the World Wide Web (WWW) to find useful information about web content, users queries, user behaviour, and structure of the web. Here users can act as consumer or contributor of its data and services. According to the present use of WWW, there is a paradigm shift from the web users from the demand of information to demand of knowledge and this requirement transfers WWW to semantic web. The semantic web is knowledge oriented and provide query relevant knowledge using clustering technique on the massive and linked data present in web on different fields. Figure 1 shows an image of open link data on the Internet (or cloud) of several domains such as social networking sites, media, publications, user generated data etc. The image shows the datasets published in the Linked Data format which currently contains 1,255 datasets with 16,174 links (as of May 2020). The figure (Figure 1) is adopted from the web page which is the home of the *LOD cloud diagram* [3].

1.1 Types of semantic web mining

Patterns discovering methods may provide tools and techniques to extract several significant contents from web by implementing data mining techniques in sophisticated manner. Web mining can be classified into three classes depending on how the web data to be mined. These are - Web Structure Mining (WSM), Web Usage Mining (WUM) and Web Content Mining (WCM) [6]. WSM techniques are concerned with the correlation that exists among related pages, WUM techniques focus on discovering the patterns of raw usage of data, while WCM refers to finding, extracting and gathering of valuable information that better suited to search query and gathering the related web pages to coherent groups. Web structure mining and content mining are often performed together which allows to exploit the hypertext content and the structure simultaneously.

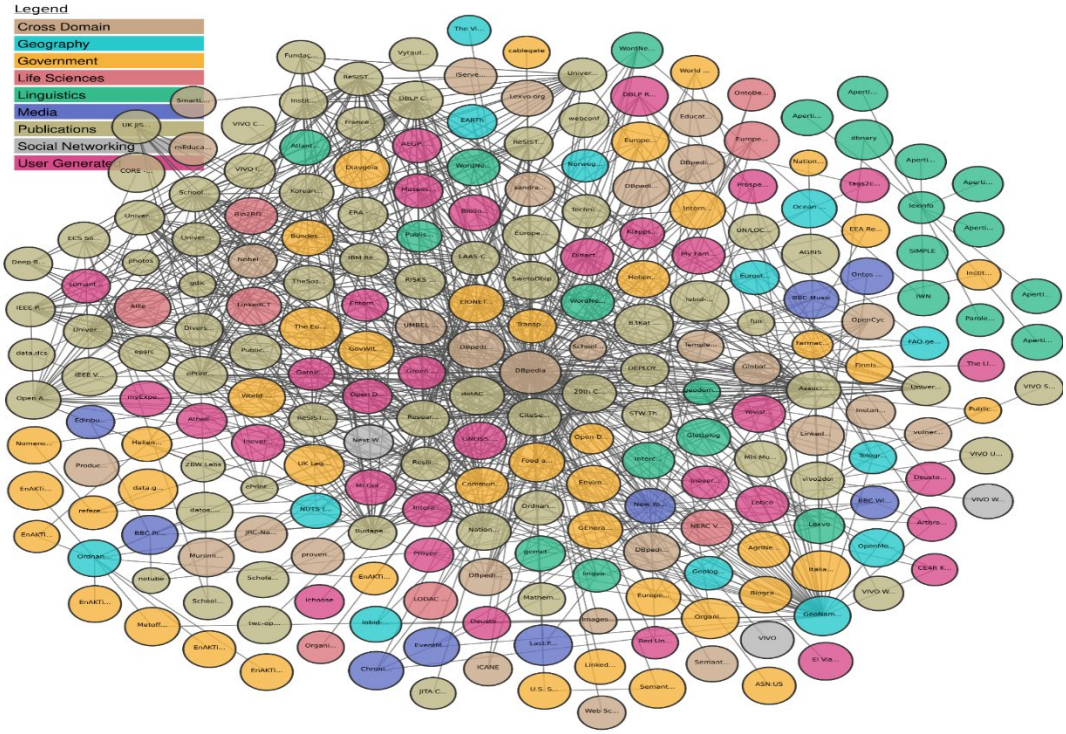

Figure 1. Linked open data cloud from lod-cloud.net

1.2 Ontology and semantic web

The semantic web has come into field as a solution to the information overload problem due to massive and ever-growing data in web. It is a machine-readable web, designed as a global document repository, with easy routes to access, publish, and link documents. Ontology is a recognized approach for knowledge representation and sharing across several applications. Ontology is the backbone of the

semantic web as it aims to deal with the structured data to develop standards and technologies designed for both users and machines understandable. It requires construction of a swift and operative ontology method for developing an erudite knowledge-based and semantic web-based system. Figure 2 simply represents how the ontology plays as a vital input in semantic web mining and as a result a model and pattern set are received as output. Along with one or more web pages, relational database, graphs, text documents etc. are also key input in such methods [7][8].

However, setting up ontology manually is a difficult task as it is not only error prone but also time consuming and so it requires participation of domain experts. So, a better solution is construction of an automatic or semi-automatic ontology methodology. Over the past years, several research attempts have been made to build such appropriate ontologies for semantic web. But there still exists many open issues in this field.

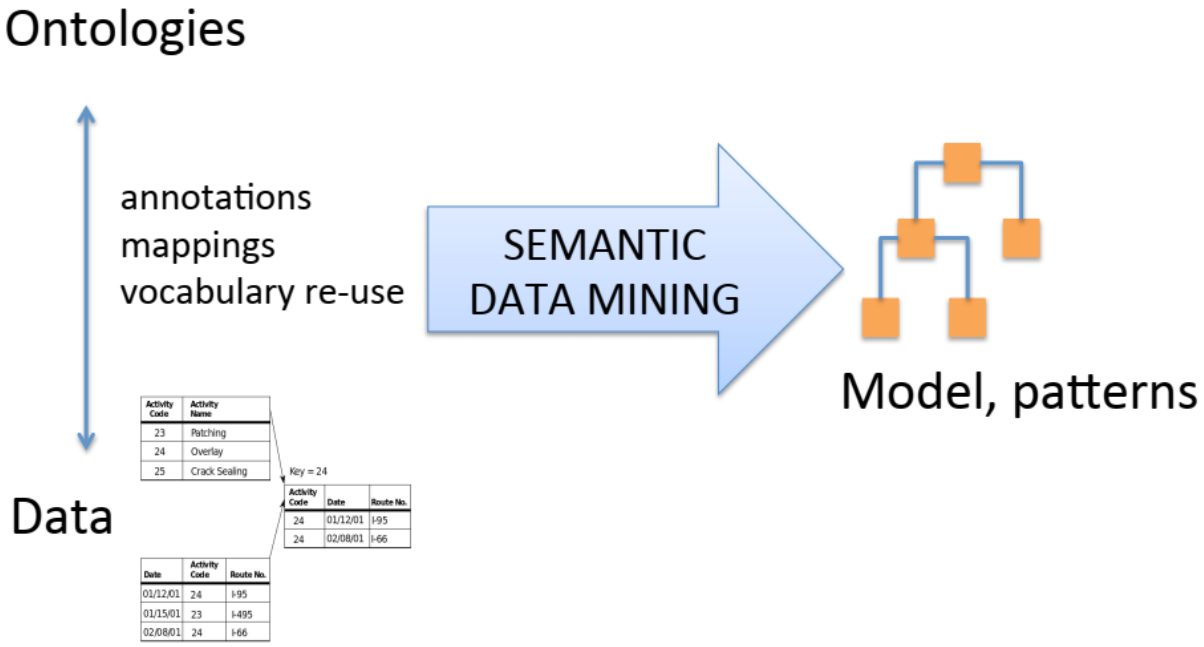

Figure 2. Ontology in semantic web data mining

The rest of the paper is organized as follows. Section 2 discuss the background of the clustering in semantic web and its importance. Then several clustering methods are thoroughly described in Section 3. The comparative results of all such methods are presented in Section 4 followed by the interpretations of the same in Section 5. The importance of clustering in SWoT applications is briefly discussed in section 6. Finally, we conclude in section 7 with some future work possibilities.

2. Background

Clustering is the method of categorizing similar object into groups, or in other words, partitioning of a dataset into subsets or groups called clusters. The process is considered as a valuable tool for semantic web agents as it is applicable in large range of problems. Clustering navigates the user to find the results in several collections of clusters relevant to the corresponding query. So, it becomes easy for the users to locate the valuable search results according to their need, thus better satisfying user requirements and providing optimal utilization of web surfing time. The main aim of semantic web pages clustering is to group together related pages depending on its contents, then this information is used to improve the results of web search engines and other applications such as information retrieval systems. New algorithms frequently required to process complex data types that are collected from several web pages for collecting meaningful contents. Among all other clustering methods, web pages clustering is a critical task due to the complex structure of web pages which is totally different from other format and combine extra embedded information.

Web page clustering is the most popular strategy in web mining that puts together web pages in groups depend on similarity or other proximity measures, where pages in the same cluster are more similar to each other than pages in other clusters [9][10]. However, the clustering is very significant and difficult task when large number of unlabeled web pages or objects frequently accessed by several users. In this regard, the new clustering algorithms should be developed, or existing algorithms should be modified efficiently to be able to propose new analysis

criteria which can match the users' requirement. However, many research works have been done so far and as a result many clustering algorithms have been proposed for different classes of web mining techniques. But this work discusses the detail of such algorithms mainly focus on web content mining and also compare their characteristics with each other. After going through the whole survey, it becomes easy to select one according to the requirements after analyzing the merits and demerits of all.

3. Categorization of web page clustering methods

In web page clustering, the data before clustering is collected in the form of web pages or search results. Then data preprocessing is done to make it suitable for clustering. During next phase, features are extracted based on web page content (mainly text-based) or interconnected web links or both content and links. Lastly, on the extracted features, clustering method is applied, and results are obtained. Here, we initially focus on the methods based on web content clustering which comes under the web content mining. Later, few clustering methods from web mining from structure and usage are also discussed [11].

Web content clustering can be defined as the form of unsupervised classification where the classes of web pages are not known previously, and which are to explore and discover the significant content from massive data by grouping the data contents into coherent points into clusters. The points that fall or exist within one cluster are similar to each other than others in different clusters. Now-a-days, most of the information on the internet are stored in form of text, this made the text mining very important topic in web mining [8]. The text-based clustering algorithms characterize every page by its contents (words or sometimes phrases are used). The main goal behind that is the pages that involve many common words, or which are likely to be very similar. Therefore, based on clustering methods, text-based approaches can be classified into the following categories – partition-based, hierarchical, graph-based, density-based and probabilistic. Furthermore, algorithms based on the way of clustering can be either hard (crisp) or soft (fuzzy). Crisp methods consider non-overlapping partitions i.e. web page either belongs to a cluster or not, while in soft approach, the page can belong to more than one cluster [12][13].

Table 1. Centroid-Based Clustering General Characteristics

Algorithm Name	Data Types	Complexity Time	Topology	Sensitive to Outliers	Input Parameter	Result
K-means [14][15]	Numerical Data	$O(n^{dk+1})$, if k, d fixed $O(nkdi)$, otherwise	Spherical	√	k, i, d, n, random state	E, K
PAM [16]	Numerical Data	$O(\ k(n-k)^2)$	Non-convex	×	A, k, d	K
CLARA [16]	Numerical Data	$O(ks^2+ k(n-k))$	Non- convex	×	s, k, i	K
CLARANS [16]	Numerical Data	$O(n^2)$	Non- convex	×	s (all neighbors of current node), n, k, i	The best local optimum result as K
FCM [17]	Numerical Data	$O(ndk^2i)$	Non- convex	×	n, c	Assigned data-values to clusters, K
KK-means [18]	Numerical Data	$O(n^2(i+d))$	Non- convex	√	kernel matrix, k	K
WK-means [18]	Numerical Data	$O(nkt)$	Non- convex	√	kernel matrix, k, weights for each point/object	K

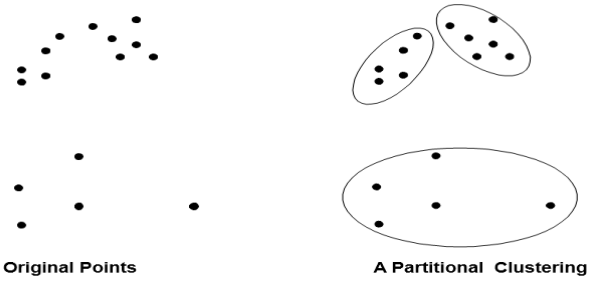

Figure 3. Spherical shapes produced from Partitioning clustering algorithms

3.1 Flat or Partition-based clustering

The partition-based clustering method classifies the information into multiple groups based on the similarity and characteristics of the data. The number of clusters that has to be generated for the clustering methods can be known from the data analysis. Given a data set (D) of multiple (n) web objects, to form k number of clusters, partitioning method constructs user-specified k partitions (where $k < n$) of the data in which each partition represents a cluster and a particular region. The most widely used partitioning methods are K-means and K-medoids. These are classical partitioning algorithms and several variations of these methods are used today in web to handle growing volume of data.

The method K-means is a centroid based technique where partitioning clustering algorithms aim to point assignment procedure. In other words, the main point, called ***Centroid C,*** is initially selected in each cluster by computing the mean value of the objects in the cluster. Then each object (among *n* multiple objects) of *D* is assigned to the cluster having closest *Centroid* value i.e. to which the object is the most similar according to the mean value. The main objective from partition-based clustering algorithms is to find k clusters represented by C_i= $\{C_1, C_2,\ldots, C_k\}$ by partitioning dataset *D* to coherent groups k. Clusters choose Centroids randomly at the initialization step. The process of partitioning is frequently repeated depending on number of trials, where at each trial the cluster centroids are updated, till the no change in *Centroid* or clustering algorithms conduct optimally similar clusters. However, the partitioning algorithms produce spherical shape clusters as they assign the data point to its closest cluster centroid [14]. Figure 3 represents the results of final clustering from the initial/original points after applying *k*-means algorithm where the spherical shape clusters produced from variant text document dataset.

In centroid based techniques, optimizing the intra-cluster variation is computationally challenging task which measures the partitioning quality of cluster C_i. For the within-cluster variation, for each object in every cluster (1,..,k), the distance between the object and its cluster centre is squared, and the distances are summed up; it can be represented as in equation 1.

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} dis(p, c'_i)^2 \quad (1)$$

where E is the sum of the squared error for all objects in data set D in d dimensional space, p is the point in space representing a given object; and c'_i is the centroid of cluster C_i and $dis(p, c'_i)$ is the distance (Euclidean distance) between data point p and centroid c'_i.

In K-medoids method, which is an object-based technique, the objects mean value is not used as a reference point in a cluster. Instead, the actual objects are used to represent the clusters, using one representative object per cluster. Then, each remaining object is assigned to the cluster according to the closest representative object i.e. whose representative object is found as the most similar. Then the partitioning process is performed following the principle of minimizing the sum of the dissimilarities between each object p and its corresponding representative object. This method groups n number of objects into k clusters thus minimizing the absolute error which can be defined as in equation 2.

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} dis(p, o_i) \quad (2)$$

Where E is the sum of the absolute error for all objects p in D, and o_i is the representative object of C_i.

Medoid is less effected by outliers or other extreme values compared to mean. Thus, in the presence of outliers and noise, K-medoids method is more robust than K-means. However, the complexity of each iteration in K-medoids increases for large values of k and n, and such computation becomes more costly than the K-means method.

The main partitioning goal is to get coherent number of clusters which have minimal Sum of Squared Error (SSE). Algorithm of K-means, Clustering Large Applications (CLARA), Partitioning Around Medoids (PAM), and extended versions of K-means are the most common algorithms used in many text-based web content clustering applications. Table 1 illustrates the general characteristics of centroid based algorithms, t=time to calculate the distance between two data objects, n=total no. of objects in dataset to be clustered in d dimensional vectors, i=no. of interactions, s= size of the random sample of the dataset chosen in the algorithm closely representing the original data, m= set of elements/medoids, A= adjacency matrix showing distance between medoids, K=set of clusters or $\{C_1, C_2,\ldots, C_k\}$.

K-means algorithm is the most superior in terms of computational time compared to other partitioning algorithms. Though K-medoids perform better for larger data sets compared to K-means. FCM generates close results to K-means but takes more time in computation. These type of clustering algorithms are efficient in terms of

lower computational complexity but has low scalability and sensitive to outliers.

3.2 Hierarchical clustering

Hierarchical clustering methods are developed to overcome the drawbacks presented in portioning or flat based clustering algorithms. Hierarchical clustering analysis can be classified into two approaches: *Divisive* and *Agglomerative* approach. In *Agglomerative*, the process starts from the cluster having single data point then integrate sub-clusters into big cluster and so on. It calculates proximity matrix i.e. similarity of one cluster with all other clusters. Then the clusters which are highly similar to each other are merged and then the proximity matrix is recomputed. The process is repeated until only one cluster remains there. The reverse of *Agglomerative* method is *Divisive*. *Divisive* method starts from the inclusive cluster and segregate the cluster points recursively into sub-clusters until certain number of similar clusters is gathered at each iteration [18]. For segregation of the cluster points at each step, any partition-based algorithm can be used. Figure 4 presents the procedure of both types of hierarchical algorithm where each alphabet is considered as a single cluster. Table 2 illustrates the characteristics of most well-known hierarchal algorithms developed in different times where $n =$ number of data points/objects as mentioned above, m_m and m_a are average and maximum number of neighbors for a point respectively, r is the cluster radius. *Divisive* algorithms are more efficient in terms of running time and accurate, but they are computationally more complex as compared to *Agglomerative* techniques. Figure 5 shows diverse shaped clusters which are produced from hierarchal algorithms [15].

Hierarchal algorithms can join with other clustering algorithms such as K-means to find coherent groups, for example clustering of text documents. Both the hierarchical and partition-based algorithms are distance-based clustering method which can be used for any data types if a proper distance function is created for that type. In such methods, the clustering quality depends on the algorithm, the distance function, and the application where inter cluster distances should be maximized and intra cluster distances should be minimized. Thus, designing of appropriate distance function is very crucial task and important area of research in web data mining. But hierarchical algorithms are typically not commonly used when data has multidimensional space due to the following reasons:

- It requires high computation time.
- Number of clusters should be known in prior.
- Continuous clustering process produces large in-coherent cluster

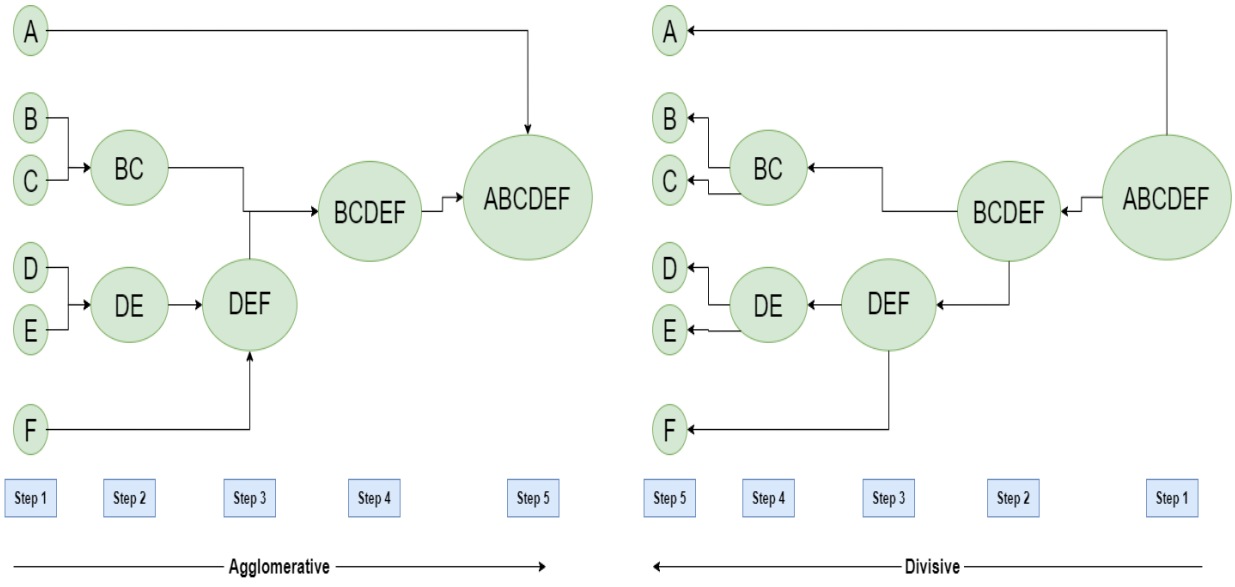

Figure 4. Hierarchical clustering method

Table 2. Properties of Hierarchical Algorithms

Algorithm name	Data types	Time Complexity	Topology	Sensitive to outliers/noise	Input parameter	Result
BIRCH [19]	Numerical	$O(n)$	Spherical	Handle noise effectively	r, branching factor	K, E
ROCK [20]	Categorical	$O(n^2 + nm_m m_a + n^2 log\ n)$	Arbitrary	×	k	Assigned data-values to clusters
CURE [19]	Numerical	$O(n^2 log\ n)$	Arbitrary	Less sensitive to noise	k	Assigned data-values to clusters
AHC[21]	Numerical	$O(n^2)$	Arbitrary	√	k	Assigned data-values to clusters
SL [21]	Numerical	$O(n^2 log\ n)$	Arbitrary	√	k	Assigned data-values to clusters
CL [21]		$O(n^3)$		Not strongly affected by outliers		

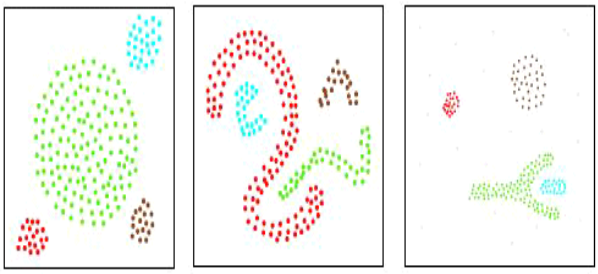

Figure 5. Clusters produced from Hierarchical algorithms

Thus, due to many irrelevant dimensions for the high dimensional data in today's web, the quality of distance function may be reduced. It may exhibit errors that in turn reduces the statistical significance of web data mining results. Moreover, the size of web clusters varies, and web datasets contain noises. Most of the hierarchical clustering methods are sensitive to outliers. *Outliers* are not assigned to any cluster and, they can be considered as anomalous points depending on the context. So, for the huge and growing web datasets and due to the arbitrary shapes of web clusters, density-based clustering is preferred over distance-based clustering methods such as hierarchical and partition-based algorithms.

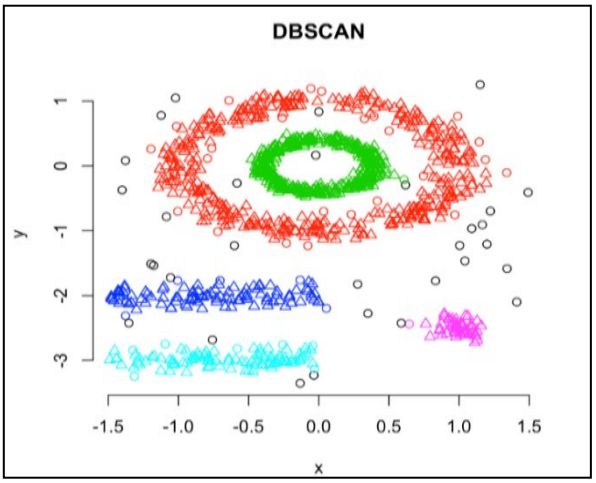

Figure 6. Clusters produced from Density based algorithm

Table 3. Most Common Density-Based Clustering Characteristics

Algorithm name	Data type	Time complexity	Topology	Sensitive to outlier/ noise	Input parameter	Result
DBSCAN [23][24]	Numeric	$O(n\ log\ n)$	Arbitrary	√	(r, Min_p)	Data values assigned to clusters
OPTICS [25]	Numeric		Arbitrary	√	(r, Min_p)	Data values assigned to clusters

DENCLUE [26]	Numeric	$O(log\|D\|)$	Arbitrary	√	(r, Min_p)	Data values assigned to clusters

3.3 Density based clustering method

Density based clustering methods typically work base on intensity of local data points to intimate clusters rather than using similarity or distance measures. It works by detecting "dense" clusters of points thus learning clusters of arbitrary shape and identifying outliers in the data. The clustering here grows incrementally while the intensity of data points in neighborhood greater than pre-defined threshold. Main objective of this kind of method is to determine the non-spherical clusters.

Figure 6 shows the results from application of most popular density-based clustering algorithm DBSCAN (Density-based spatial clustering of applications with noise) on a dataset finding clusters based on their density, as opposed to their distance from a centroid (as K-means would). DBSCAN algorithm is one of the three algorithms which are awarded the Test of Time Award at SIGKDD 2014. Unlike distance-based algorithms such K-means which generally discovers spherical clusters, DBSCAN can discover arbitrary shaped clusters and can find non-linearly separable clusters. The natural setting for the density-based algorithms is spatial data clustering as these methods require a metric space.

Table 3 shows the characteristics of the most widely used density-based clustering algorithms, where r refers to cluster radius (maximum distance to consider), and Min_p refers to the minimum number of data points needed in a neighbourhood to define a cluster. Apart from the classical density-based algorithms presented in the table, there are some algorithms under the two broad categorizations of density-based clustering methods such as – ExCC, MR-Stream etc. under density grid-based clustering and Denstream, FlockStream etc. under density micro clustering method. These clustering methods have better quality of clusters than grid-based methods, but they need more computation time. So, later different hybrid methods proposed modifying DBSCAN algorithm, but they are not suitable for today's distributed web environments and web contents [22].

In these methods, clustering results are sensitive to parameters and for a large volume of data, huge memory is needed. When the data space density is uneven, the method results in low quality clusters.

3.4 Graph based clustering

Graph based algorithms initially construct a graph or hyper-graph then apply clustering algorithm to partition the graph result. For clustering the web contents, the web pages can be viewed as a set of nodes and the web links are the edges among nodes representing the strength of relationship. A set of vertices is considered as a good cluster if it has low conductance i.e., if it has more external edges than internal. However, drawbacks of such algorithms are as follows - the graph must fit in the memory, and the technique that is used for calculating similarity among nodes have to use cut-off property. This type of clustering is a hybrid method using content and link both for feature extraction (Section 2) and comes under the web mining based on structure (WSM). Figure 7 shows a simple example of graph representation of linked web documents/pages with text (TX), title (TI), link (L) [27].

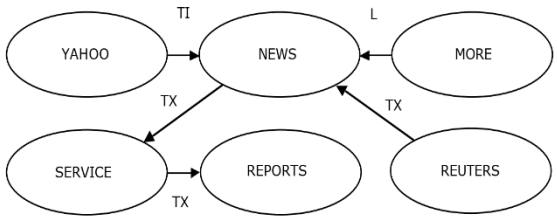

Figure 7. Graph representation of web content

Graph construction methods extract a similarity graph which conserves the key properties of the dataset and to do so, they involve a sparsification of the similarity matrix under different heuristics (from simple thresholding to sophisticated regularisations). However, the choice of sparsity (method parameters) has a strong impact on the performance of such methods. The workflow in graph-based clustering method is shown in Figure 8.

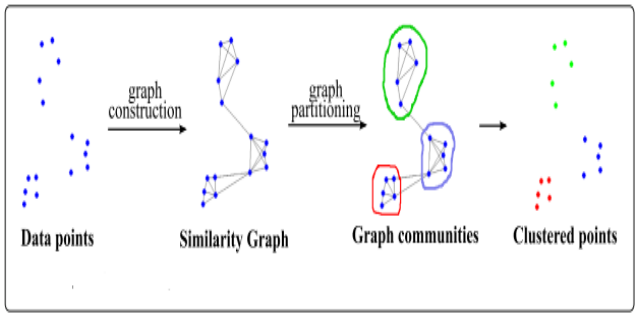

Figure 8. Basic idea of Graph-based clustering method

Many literatures have conjunction with the graph partitioning and community proposed many methods for graph construction in detections from high dimensional dataset such as web data. Among different graph construction algorithms, in this regard, the most commonly

used methods are $\in$ −ball graph, kNN, CkNN etc. for graph construction and Markov Stability (MS) for community detection [28]. MS has successfully applied in social networks, airport networks etc. Other dynamical processes have also been applied widely in network analysis. The high dimensional nature of web data leads to complex geometries associated with datasets and thus it poses challenges to standard clustering methods. The use of graph clustering method in this regard helps in capturing complex geometry of dataset and managing complex network analysis in web.

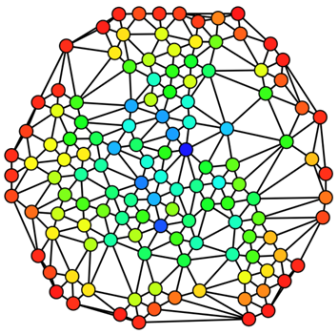

Figure 9. Clusters generated from Graph clustering method on large web data

Figure 9 shows an example of clusters produced from graph-based algorithms on big data. There are several methods to represent content of a web document as graphs such as *standard*, *simple*, *n-distance*, *absolute* and *relative frequency* etc. Each method looks for runs of characters separated by blank spaces or other common punctuation marks on each web page, their adjacency and then extracts the terms. Once the terms on the web pages are extracted, several steps are applied to reduce the number of terms omitting the irrelevant ones so that computation time of clustering decreases.

Table 4 illustrates the most common characteristics of graph-based algorithms where n number of objects each with m number of attributes to be clustered, E= number of links or edges, v= starting vertex, φ=target conductance, h=size of the sparsity of given dataset D and graph G, C is any set with small conductance, and the resulting PageRank vector is not close to the stationary distribution for many starting vertices contained in C, as it has significantly more probability within C. The problems in this kind of clustering methods for web mining are selecting/determining the input parameters, memory consumption and runtime for massive input data such as in today's web. In addition, it is also difficult task to select the appropriate method for clustering with the massive growth of web resources.

Graph based clustering methods result in high quality and accurate clusters but as the complexity of the graphs increases, the time complexity of the algorithms increases drastically.

3.5 Probabilistic Clustering

Probabilistic clustering, also called distribution-based clustering, is a special type of hard clustering method. The probabilistic clustering algorithms are most closely related to statistics that follow Bayesian classification arguments. In such methods, the data is considered as a sample which is independently drawn from a mixture model of several probability distributions. Here, each vector x is assigned to the cluster C_i for which $Probability(C_i \mid x)$ is maximum, i.e. where a vector belongs to a specific cluster. The assignment of the vectors to individual clusters is carried out optimally, according to the optimality criterion. These methods are extensively used in many applications such as recognition of handwriting, clustering text document, retrieval systems and topic modeling. Figure 10 shows a scatter plot of dataset with clusters identified using Gaussian Mixture clustering by python.

These algorithms use statistical models rather than predefined similarity measures to calculate the similarity among data points. But, the time complexity of these algorithms is quite high, and they converge slowly in some situation. So, clustering huge amount of web data today where time is the prime factor, these methods are not always suitable. Table 5 shows the most widely used probabilistic clustering algorithms and their characteristics where n number of objects each with m number of attributes to be clustered, k is the number of **Gaussian** components/clusters here and data are produced from the mixture of k distributions $G_1,\ldots, G_k$, and distribution parameters are mean, variance etc.

Table 4. Graph-Based Algorithms Common Properties

Algorithm Name	Data Types	Time Complexity	Topology	Outlier	Input Parameter	Result
DIG[29]	Numeric	$O(mm^2)$	Graph	√	C, Cut threshold, No. of cells	Hypergraph
ICA[30]	Numeric	$O(\lvert E\rvert + h \log h)$	Graph	√	C, Cut threshold, No. of cells	bipartite graph

Nibble [31]	Numeric	$O(2^b \frac{log^6 m}{\varphi^4})$	Graph	√	C^*, v, conductance, Cut threshold	Hypergraph
Page Rank Nibble[32]	Numeric	$O(2^b \frac{log^4 m}{\varphi^5})$	Graph	√	Φ, v, integer $b \in [1, log\ m]$	Hypergraph
GHCA[28]	Numeric	$O(n^2 m)$	Graph	√	Maximum Terms Threshold, Minimum Pages Threshold, Maximum Distance Threshold, Maximum Cluster Threshold, Base Cluster Size Threshold	bipartite graph

Table 5. Characteristics of common Probabilistic Clustering Algorithms

Algorithm Name	Types of data	Time complexity	Graph Topology	Outlier	Input parameter	Result
EM[33]	Numeric	$O(mn^3)$	spherical	√	likelihood of the training data/objective function, distribution parameters	Assign data points to clusters based on Gaussian probability distribution
PLSA[34]	Numeric	$O(kn)$	Arbitrary	√	k, m, n	Assign data points to clusters from Probability Density Function values
Auto Class [35]	Numeric	$O(n^2)$	Arbitrary	√	k, D	Class membership values
Gaussian mixture model [33]	Numeric	$O(nkD^3)$	Arbitrary	Less sensitive	prior probabilities, $G_1, \ldots, G_k$, distribution parameters	assignment of data points to clusters according to probability distribution

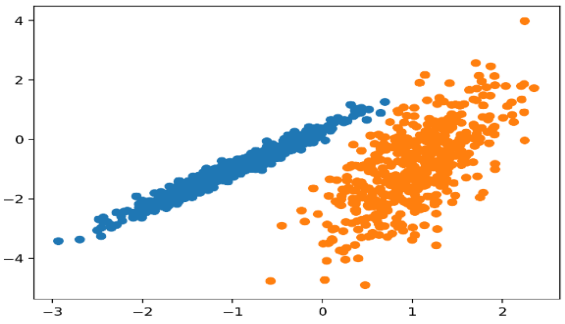

Figure 10. Clusters generated using Gaussian Mixture

3.6 Special clustering methods

Apart from the clustering methods there are some special clustering which do not exactly fit in any of the above categories. Here we briefly discuss few of them.

3.6.1 Branch and bound

This method exploits the dynamic programming principle. It provides globally optimal clustering without need of considering all possible clustering based on certain prespecified conditions (for certain fixed number of clusters). A variant of this method, called A* search is used in some applications in today's web.

3.6.2 Stochastic

Like Branch and bound method, based on certain prespecified condition, stochastic method guarantees convergence in probability to the globally optimum clustering.

3.6.3 Genetic

Based on certain prespecified conditions, Genetic clustering generates new population of clustering at each iteration using an initial population of possible ones.
Apart from the above method there are some clustering methods proposed in different times [35][36] which does not fall in any specific category of clustering, rather, may be considered as a mixed clustering technique. These

methods intend to use more than one standard clustering techniques.

4. Results and Discussion

Most of well-known clustering algorithms have many limitations and drawbacks. The methods are not able to work properly with ever-growing web data till we modify such parameters which adjective these algorithms with web data. Additionally, one of the most critical challenges is intimated here when text data that comes from web environment do not have labeling property. However, this challenge has made the evaluation process of any clustering results more difficult task in which depends on analysis model and validation measures. This paper aims to revise the studies and perceptions looking for the algorithms that are mostly used with text content in the web.

In table 6, a comparative study of all such techniques discussed here is presented based on the real time application, CPU time used, memory consumption, algorithm limitations, and data dimensionality. The terms used for comparisons are follow: Flexible, Low, High, Convenient, Inconvenient and Not assigned.

5. Interpretation and Evaluation

Cluster analysis of web content is significant process to understand and interpret; in this regard, choosing a suitable clustering algorithm is more difficult. This process is typically restricted based on the data types used and application purpose. The traditional clustering algorithms are unable to perform satisfactory in the current scenarios of web data mining due to several reasons such as follows.

- Scaling issues due to large number of samples to be processed
- The number of features is too large and sometimes exceed the number of samples due to high dimensionality of data
- Finding the outliers are very significant which is difficult for large volume of data
- The knowledge of previous cluster analysis can be reused to avoid starting the analysis from scratch, but they are often available
- Heterogeneous and distributed data sources where local cluster analysis results are to be integrated into global models

Most of clustering algorithm that are used with web pages consider only the text part and most of them use statistically based Vector Space Model (VSM) [36]. According to such model, a web document is represented conceptually by a vector of keywords mined from it. But none of the methods of this model is well suited for all types of queries generated in web. In the field of information retrieval, Web clustering Engines e.g. Clusty, Lingo3G, Grokker, KartOO, CREDO etc. are emerging trend which organize search results by topics [37]. They group the search results into different (hierarchical) clusters and display those cluster labels. As a result, the user can conveniently and quickly locate the desired document. This clustering includes constantly changing billions of pages. The dynamicity nature of the data along with the interactive use of the clustered results stance new needs and challenges to clustering technology such as selection of similarity measure, meaningful cluster labels, handling cluster overlapping, removing clustering ambiguity, increasing computational efficiency etc. Depending on the specific algorithm used, the clustering phase can significantly contribute to the overall processing time.

To improve the cluster efficiency, the extracted features from cluster should be powerful. In this regard, to increase the efficiency of clustering phase, the developers should adopt methods to generate more expressive and effective descriptions of clusters and should find optimal cluster representatives. But the usage of WWW is increasing everyday now and as a result in the big data era and digital world, managing the massive content in internet is becoming more difficult task. So, before choosing any clustering technique, the innovative usage and requirement of web content clustering should be realized first which can be identified as follows – classifying network traffic, identifying fake news, spam filtering, sales/marketing, analysing document.

Table 6. Comparison of Clustering Algorithms

Category	Algorithm Name	CPU Time	Memory	Suitability for large dataset	Restriction and sensitivity
Partitioning based algorithms	K-means	Flexible	Flexible	Convenient	Cluster shape, outlier
	PAM	Flexible	Low	Inconvenient	
	CLARA	Low	Low	Inconvenient	
	CLARANS	High	Flexible	Inconvenient	
	FCM	Low	Flexible	Inconvenient	
	WK-means	Flexible	Flexible	Inconvenient	
Hierarchical algorithms	BIRCH	Flexible	Low	Inconvenient	Cluster shape

	ROCK	High	Flexible	Inconvenient	Cluster shape
	CURE	High	High	Convenient	Interconnectivity ignored
	Chameleon	Flexible	High	Inconvenient	Not assigned
	SL	High	High	Inconvenient	Cluster shape
	CL				
Density-based algorithms	DBSCAN	Flexible	Flexible	Inconvenient	Cluster radius, minimum number of data objects
	OPTICS	Flexible	Flexible	Convenient	Ordering cluster time
	DENCLUE	Low	Low	Convenient	Density parameter, noise threshold
Graph-based algorithms	Page Rank Nibble	Low	Flexible	Convenient	Not assigned
	DIG	High	High	Inconvenient	Cut threshold, dataset size
	ICA	High	High	Convenient	Cut threshold, dataset size
Probabilistic algorithms	EM	High	High	Inconvenient	Missing data, dimensionality
	PLSA	High	High	Inconvenient	Dataset size
	Gaussian Mixture	Low	Flexible	Convenient	No uncertainty measure or probability to say how much a data point is associated with a cluster
	Auto class	not assign	Low	Inconvenient	Number of training samples
Other algorithms	SOM	High	Low	Inconvenient	Fixed output nodes, limit interpretation result
	STC	Low	Flexible	Inconvenient	Snippets noise, sequence of words

There are some popular clustering algorithms preferred by data scientists to gain some valuable insights from the data by examining in what group the data points fall into while applying a specific method. These are – K-means, Mean shift, DBSCAN, Gaussian Mixture Models and Agglomerative Hierarchical Clustering [38]. However, each algorithm has its own advantages and demerits, as discussed in this article, and cannot work for all real situations. So, combination of existing clustering algorithms should be used for getting better clusters.

6. Semantic web clustering in IoT applications

Today enormous number of data in different formats are generated in web from a huge number of heterogeneous devices, several networks, applications, communication protocols. With these growing number of devices, the reality of Internet of Things (IoT) and their diversity is stimulating the current technologies for a smarter integration of their applications, data and services. While the web is considered as a convenient platform to integrate the things, the semantic web on the other hand can further extend its capacity to recognize the things' data and simplify their interoperability. In this regard, the Semantic Web of Things (SWoT) is proposed for integrating the semantic web on IoT. Web of Things (WoT) allows the different things and systems to communicate together through API over HTTP or CoAP protocol. Whereas, the SWoT is the fusion of IoT trends for moving toward the web technologies with protocols like CoAP, REST architecture and WoT concept. There are semantic web technologies as well as some of the well accepted ontologies which are used to develop applications and services for the IoT. But the existing approaches are lacking behind in well-defined standards and conventional tools to solve the semantic interoperability problem in IoT applications [39][40].

One strategy to deal with these challenges is to reduce the number of discovered services using different methodologies such as clustering of semantic web. However, most of the existing approaches are suitable for static context and don't take into consideration the dynamicity of services and gateways. So, the unsupervised clustering mechanisms need to be discussed and explored with much more attention for performing analysis on IoT sensor data along with dynamicity of WoT services.

7. Conclusion

This paper aims to provides a rigorous survey of several algorithms that is used with web content clustering along with a brief discussion on the importance of semantic web clustering for WoT services. Behavior of text content in web pages is very different from the traditional text documents. So, selecting the most suitable algorithm for handling the variety in text tokens is very difficult as it depends on the domain complexity and types of data used in web. Several clustering algorithms proposed so far but the main problem with such algorithms is that they cannot be standardized.

One algorithm may give appropriate results with one type of dataset but may provide poor results with dataset of other types. Although there have been many attempts for standardizing the algorithms, but no major achievement has

been accomplished till now. Our future works aim to find the behaviour of using hybrid algorithms by successfully combined few of the popular existing algorithms such as graph-based algorithms with hierarchal algorithms etc. The aim is to able to successfully handle the heterogeneity in web content from the large datasets and heterogeneous devices and which can also be successfully implemented in SWoT applications.

References

[1] Jose Aguilar, J. (2009). A Web Mining System. Wseas Transactions on Information Science and Applications, 9(6), 1523-1532.

[2] Singh, S. and Aswal, M. S (2018). Semantic Web Mining: Survey and Analysis. Journal of Web Engineering & Technology, 5(3), 20-31.

[3] A Linked Open Data Cloud. https://lod-cloud.net/#about

[4] Sharma, K. Shrivastava, G. and Kumar, V. (2011). Web Mining Today and Tomorrow. In: 3rd International Conference on Electronics Computer Technology (ICECT), 399-403, IEEE, India. Doi: 10.1109/ICECTECH.2011.5941631.

[5] Yadav, M. and Mittal, P. (2013). Web Mining: An Introduction. International Journal of Computer Science and Software Engineering, 3(3), 683-688.

[6] Singh A. (2012). Agent Based Framework for Semantic Web Content Mining. International Journal of Advancements in Technology, 3(2), 108-113.

[7] Dou, D., Wang, H. and Liu, H. (2015). Semantic data mining: A survey of ontology-based approaches. In: 9th International Conference on Semantic Computing (ICSC), 244-251, IEEE, USA. Doi: 10.1109/ICOSC.2015.7050814.

[8] Lawrynowicz, A.(2016). Semantic Data Mining: an Ontology Based Approach, 1-66.

[9] Patel, D. and Zaveri, M. (2011). A Review on Web Pages Clustering Techniques. In: Trends in Network and Communications. WeST 2011, NeCoM 2011, WiMoN 2011. Communications in Computer and Information Science, vol. 197, 700-710, Springer, Berlin, Heidelberg. Zhang, Z., Zhao, J. and Yan, X. (2018). A Web Page Clustering Method Based on Formal Concept Analysis. MDPI Information Journal, 9(228), 1-14.

[10] Aggarwal, C. (Ed.), Reddy, C. (Ed.). (2014). Data Clustering. New York: Chapman and Hall/CRC,

[11] Lei, Y. (2017). Clustering algorithm–based fault diagnosis. Intelligent Fault Diagnosis and Remaining Useful Life Prediction of Rotating Machinery, Science Direct, 175-229.

[12] Xu, R. and Wunsch, D. (2005). Survey of Clustering Algorithms. IEEE Transactions on Neural Networks, 16(3), 645-678.

[13] K-means clustering, https://en.wikipedia.org/wiki/K-means_clustering

[14] Chawla, S. and Gionis, A. (2013). K-means–: A unified approach to clustering and outlier detection. In: SIAM International Conference on Data Mining, 189-197, Society for Industrial and Applied Mathematics.

[15] Schubert, E., Rousseeuw, P. J. (2019). Faster *k*-Medoids Clustering: Improving the PAM, CLARA, and CLARANS Algorithms. In: International Conference on Similarity Search and Applications (SISAP), Lecture Notes in Computer Science, 11807, 171-187, Springer, USA.

[16] Bora, D. J and Gupta, A. K (2014). A Comparative study Between Fuzzy Clustering Algorithm and Hard Clustering Algorithm. International Journal of Computer Trends and Technology, 10(2), 108-113.

[17] Dhillon, I. S, Guan, Y., Kulis, B. (2004). Kernel K-Means: Spectral Clustering and Normalized Cuts. In: 10th SIGKDD International Conference on Knowledge Discovery and Data Mining, 551-556, ACM, USA.

[18] Firdaus, S. and Uddin, M. A (2015). A Survey on Clustering Algorithms and Complexity Analysis. International Journal of Computer Science Issues. 12(2), 62-85.

[19] Guha, S., Rastogi, R. and Shim, K. (2007). ROCK: A Robust Clustering Algorithm for Categorical Attributes, 1-48.

[20] Rafsanjani, M. K. and Varzaneh, Z. A (2012). A survey of hierarchical clustering algorithms. The Journal of Mathematics and Computer Science, 2012, 5(3), 229-240.

[21] Amini, A., Saboohi, H., Wah, T. H and Herawan, T. (2014). A Fast Density-Based Clustering Algorithm for Real-Time Internet of Things Stream, Hindwai Scientific World Journal, 2014, 1-14.

[22] DBSCAN, https://en.wikipedia.org/wiki/DBSCAN.

[23] OPTICS, https://en.wikipedia.org/wiki/OPTICS_algorithm.

[24] Liu, Y., Liu, D., Yu, F. and Ma, Z. (2020). A Double-Density Clustering Method Based on "Nearest to First in" Strategy. MDPI Symmetry Journal, 12(747), 1-18.

[25] Nagpal, P.B. and Mann, P. A. (2011). Comparative Study of Density based Clustering Algorithms. International Journal of Computer Applications, 27(11), 44-47.

[26] Liu, Z. and Barahona, M. (2020). Graph-based data clustering via multiscale community detection. Applied Network Science, Springer, 5(3), 1-20.

[27] Schenker, A. (2003). Graph-Theoretic Techniques for Web Content Mining. Graduate Theses and Dissertations, University of South Florida,https://scholarcommons.usf.edu/cgi/viewcontent.cgi?article=2466&context=etd;Graph-theoretic.

[28] Hammouda, K. M and Kamel, M. S. (2004). Efficient Phrase-Based Document Indexing for Web Document Clustering. IEEE Transactions on Knowledge and Data Engineering, 16(10), 1279-1292.

[29] Rege, M., Dong, M. and Fotouhi, F. (2008). Bipartite isoperimetric graph partitioning for data co-clustering, Data Mining and Knowledge Discovery, 16(3), 276-312.

[30] Spielman, D. A and Teng, S. H. (2013). A Local Clustering Algorithm for Massive Graphs and Its Application to Nearly Linear Time Graph Partitioning. SIAM Journal on Computing, 42(1), 1-26.

[31] Andersen, R., Chung, F. and Lang, K. (2006). Local Graph Partitioning using PageRank Vectors. In: *47th Annual IEEE* Symposium on Foundations of Computer Science (FOCS'06), 475-486, Berkeley, CA.

[32] Berkhin, P. (2006). A Survey of Clustering Data Mining Techniques. In: Grouping Multidimensional Data, 25-71, Springer, Berlin.

[33] Kuta, M. and Kitowski, J. (2014). Comparison of Latent Semantic Analysis and Probabilistic Latent Semantic Analysis for Documents Clustering. Computing and Informatics, 33, 652-666.

[34] Cheeseman, P.C and Stutz, J. C. (1996). Bayesian Classification (AutoClass): Theory and Results. Advances in Knowledge Discovery and Data Mining, AAAI Press/MIT Press.
[35] Raut, A. B. and Bamnote, G. R. (2012). Vector Space Model in Clustering Web Documents. International Journal of Advanced Research in Computer Science, 3(3), 706-709.
[36] Theresa, D. (2011). Web Clustering Engine, Thesis, 1-33, Kochin University of Science and Technology. https://www.scribd.com/document/234322006/Deepthi-Webclustering-Report.
[37] Clustering Algorithms with Python, file:///C:/Users/Dell/Documents/Web%20mining%20paper%20for%20book%20chapter/10%20Clustering%20Algorithms%20With%20Python.html
[38] Nadim, I., Elghayam, Y. and Sadiq, A. (2018). Semantic discovery architecture for dynamic environments of Web of Things. In: International Conference on Advanced Communication Technologies and Networking (CommNet), pp. 1-6, IEEE, Morocco.
[39] Jara, A. J, Olivieri, A. C., Bocchi, Y. and Jung, M. (2014). Semantic Web of Things: an analysis of the application semantics for the IoT moving towards the IoT convergence. International Journal of Web and Grid Services, 10(2), 244-272.