Fuzzy Clustering and Visualization of Information for Web Search Results

Faraz Zaidi

Fuzzy Clustering and Visualization of Information for Web Search Results

FARAZ ZAIDI
Karachi Institute of Economics and Technology, PAF Base, Korangi Creek, Karachi-75190, Pakistan
faraz@pafkiet.edu.pk

Abstract

Searching for information on the web is a common task. Often information on the web is distributed, semi-structured, overlapping and heterogeneous. Organization and Visualization of this information is an active area of research where the goal is to help users locate required information in web pages efficiently.

The most widely used data organization technique is clustering. This paper introduces a new clustering algorithm to organize web pages, and a visualization method which facilitates users to search information efficiently from the web. The algorithm presented is a hierarchical fuzzy clustering algorithm which uses domain knowledge to determine input parameters as opposed to other existing algorithms in the literature. The comparative results show that the algorithm performs as well as existing algorithms. Next, we present a methodology to visualize the clustered collection of documents and their contents such that users can visually explore data and extract information. A detailed example is presented to demonstrate various views to visualize clusters, documents and the keywords present in the web pages.

Keywords: Web Mining; Information Retrieval; Fuzzy Hierarchical Clustering; Visualization

1 Introduction

The explosion of web pages has created an overload of information available on the Internet. Often this information is distributed, semi-structured, overlapping and heterogeneous [2]. Development of efficient methods to organize and visualize this information is an active area of research [8] where the goal is to help users access this information quickly. A typical solution to this problem in the context of information retrieval from the web is to organize and to visualize search results of a query launched on a search engine [26]. Search engines such as Google; tend to return a long list of search results with titles, small images and short paragraphs. Users have to open each and every web page to assess its utility and relevance to the searched topic which can become tedious and unproductive [21].

As an example, consider searching for the word Jaguar using Google Search Engine. Top seven results returned by Google (see Figure 2), are distributed heterogeneously in the list, i.e. pages 1, 2, 3 and 6 are about the automobile Jaguar, pages 5 and 7 are about jaguar the animal and page 4 is about a super computer called Jaguar. If we look further down in the list, we find pages related to a software solution provider, a musical group, and a guitar manufacturing company all having the name Jaguar. Ideally we would like to group the pages together based on their content so that the user can immediately realize that there are multiple topics or themes related to the searched keywords as shown in Figure 1. To represent the contents of these groups, displaying noun phrases and keywords would allow users to glance contents of search results without reading or scanning individual web pages and thus reduce their effort in locating relevant information [36].

Clustering has long played a pivotal role in the organization of information and has been used by several researchers efficiently for information retrieval [29]. In the context of web pages, clustering has several applications such as organization of web search results [21], web browsing [39] and automated categorization of web pages [15].

Typically, document clustering techniques consider documents as entities, and calculate a similarity between documents based on keywords appearing in these documents. This similarity is further used to group similar documents together to obtain clusters. Intuitively, an alternative approach is to consider keywords as entities related to each other if they appear in a single document. Clustering keywords can group Themes together such that when a keyword is searched, its theme can be identified from the cluster it belongs to. Obviously documents can be associated to these themes and returned as search result to the user. We discuss a number of issues that need to be addressed before selecting the right clustering algorithm.

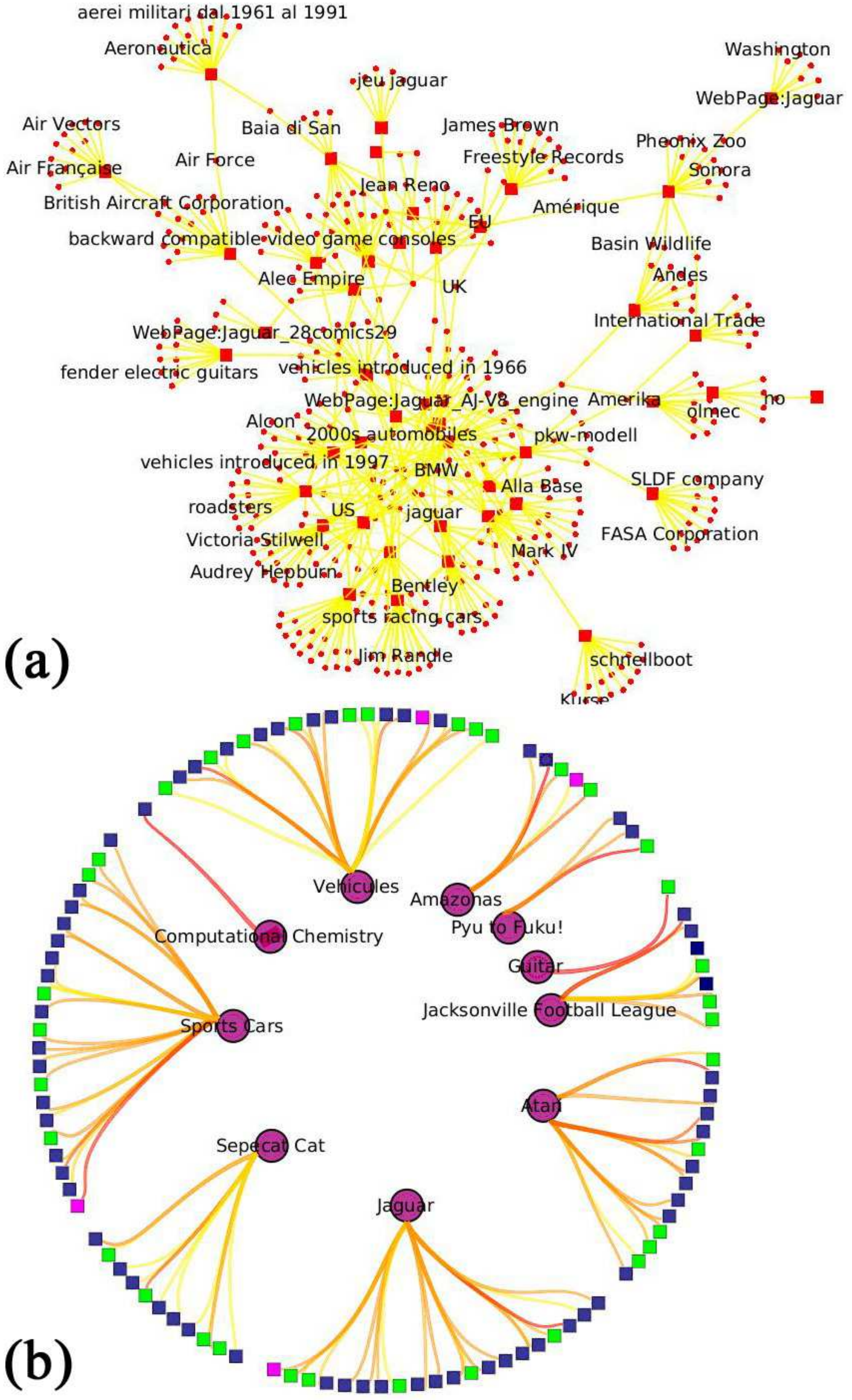
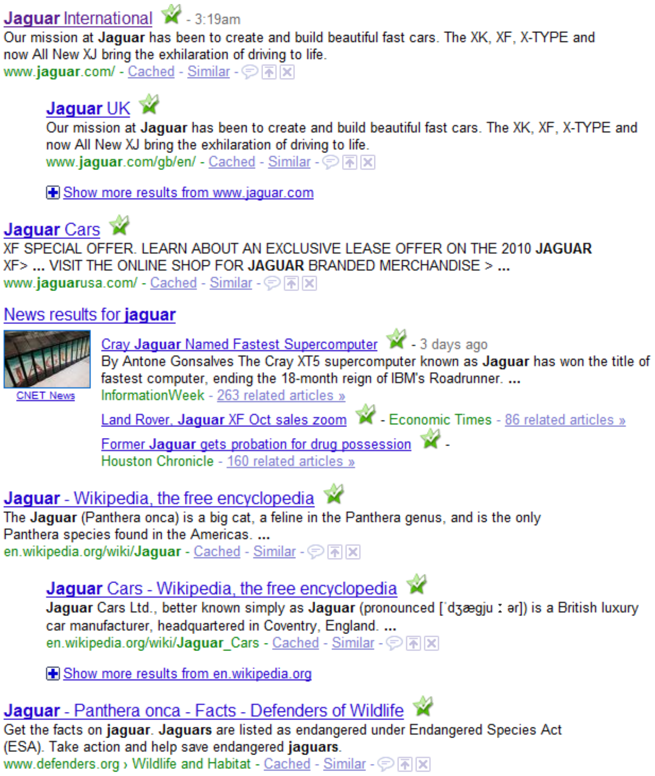


Figure 1. (a) Showing the Bi-partite graph of Web pages and Keywords (b) Clustered Network showing Clusters and Web pages.

An important issue is whether the clustering algorithm should generate *Hard* clusters or *Soft* clusters, i.e., documents (or web pages) should belong to unique clusters or may be associated to more than one cluster. Researchers [2, 9, 37] have shown that very often, a web page can belong to more than one category and thus it is more useful to use Soft or Fuzzy clustering algorithms. A decision is also required to choose between Hierarchical or Flat clustering. Naturally, information around us is organized in a hierarchical manner. Again this claim is supported by many researchers that tend to organize documents and web pages in hierarchies [11, 32, 43].

Having said that we need hierarchical and fuzzy clustering algorithms to cluster web pages, we would like to point out some other requirements induced by the domain, that is, the web. In terms of hierarchical clustering, generating a hierarchy of high depth is not suitable, as the famous Three-Click Rule [41] suggests, users tend to abandon a site if they don't find their required information within three clicks. Therefore, the depth of the hierarchical clusters should not be more than 3 levels deep making the navigation hierarchy shallow and wide.



Jaguar International - 3:19am
Our mission at **Jaguar** has been to create and build beautiful fast cars. The XK, XF, X-TYPE and now All New XJ bring the exhilaration of driving to life.
www.**jaguar**.com/ - Cached - Similar -

Jaguar UK
Our mission at **Jaguar** has been to create and build beautiful fast cars. The XK, XF, X-TYPE and now All New XJ bring the exhilaration of driving to life.
www.**jaguar**.com/gb/en/ - Cached - Similar -

Show more results from www.jaguar.com

Jaguar Cars
XF SPECIAL OFFER. LEARN ABOUT AN EXCLUSIVE LEASE OFFER ON THE 2010 **JAGUAR** XF> ... VISIT THE ONLINE SHOP FOR **JAGUAR** BRANDED MERCHANDISE > ...
www.**jaguar**usa.com/ - Cached - Similar -

News results for **jaguar**

Cray **Jaguar** Named Fastest Supercomputer - 3 days ago
By Antone Gonsalves The Cray XT5 supercomputer known as **Jaguar** has won the title of fastest computer, ending the 18-month reign of IBM's Roadrunner. ...
InformationWeek - 263 related articles »

CNET News

Land Rover, **Jaguar** XF Oct sales zoom - Economic Times - 86 related articles »

Former **Jaguar** gets probation for drug possession -
Houston Chronicle - 160 related articles »

Jaguar - Wikipedia, the free encyclopedia
The **Jaguar** (Panthera onca) is a big cat, a feline in the Panthera genus, and is the only Panthera species found in the Americas. ...
en.wikipedia.org/wiki/**Jaguar** - Cached - Similar -

Jaguar Cars - Wikipedia, the free encyclopedia
Jaguar Cars Ltd., better known simply as **Jaguar** (pronounced [ˈdʒægju ː ər]) is a British luxury car manufacturer, headquartered in Coventry, England. ...
en.wikipedia.org/wiki/**Jaguar**_Cars - Cached - Similar -

Show more results from en.wikipedia.org

Jaguar - Panthera onca - Facts - Defenders of Wildlife
Get the facts on **jaguar**. **Jaguars** are listed as endangered under Endangered Species Act (ESA). Take action and help save endangered **jaguars**.
www.defenders.org › Wildlife and Habitat - Cached - Similar -

Figure 2. Screen Shot of the top seven Search Results returned by Google for the searched term Jaguar.

Another requirement from the fuzzy logic perspective is that once we have calculated the degree of similarity of a web page to various clusters, we need to find a threshold which assures that only relevant pages are grouped together. The correct threshold value depends on the collection of web pages, the words searched and the degree of semantic disambiguity in the searched keyword. To solve this complex problem, instead of forcing a threshold value, we propose a method to visualize documents and clusters that allows a user to determine the relevance of a document to a cluster by examining its contents.

Finally the classical problem of deciding the number of clusters to be generated is also important. Instead of using it as an input parameter, this number should be determined based on the data itself. Also, in case of a hierarchical algorithm, the number of clusters generated at each level needs to be considered as well.

In this paper, we take a different approach to solve all these issues by looking at the co-occurrence network of keywords. These keywords are extracted from web pages to represent the contents of a webpage. A co-occurrence network is a graph where the nodes are represented by keywords and edges between keywords means that they appear together at least once in a web page. These networks have some interesting properties that can be used to devise heuristics which can eventually help us resolve issues described earlier. Based on these properties, we propose a new Hierarchical Fuzzy Clustering Algorithm based on Co-Occurrence Networks (HFC-CN) where the algorithm automatically determines the initial centroids and in turn, the number of clusters. The performance of the proposed algorithm is compared with other existing algorithms and the results are satisfactory as shown in Table 1. Note that we do not claim that the proposed algorithm outperforms the existing algorithms, but the automated parameter selection performs as well as the case where these parameters are selected manually. Next we use graph drawing algorithms to visually represent clusters, documents and their keywords (see Figure 1) to help users retrieve information. Details of the visualization methods are presented in Section 7.

The paper is organized as follows: In the following section, we summarize recent work related to Document Clustering using the Hierarchical Fuzzy Clustering Algorithms and Document Visualization techniques for web pages. Next, in Section 3, we formalize the problem and explain how the co-occurrence networks are constructed. Section 4 presents the data sets used as examples for experimentation. In Section 5, we present the details of the proposed clustering algorithm. Section 6 presents the results of the experimentation followed by discussion. Next, we present the visualization methodology using a case study in section 7. In the end, we present our conclusions and future research prospects in Section 8.

2 Related Work

2.1 Clustering

There are a number of document clustering algorithms available in the literature. We briefly present the different taxonomies of clustering algorithms in the context of document clustering and then focus on hierarchical fuzzy clustering algorithms.

Clustering algorithms can be classified by different criteria. One classification can be based on the resulting clusters which can either be hierarchical or partitional [18]. Hierarchical algorithms can be further classified as agglomerative or divisive. Another way to classify algorithms is the way in which similarity measure is being calculated [38] between documents like single link, complete link and the average link. We can also classify the algorithms if the resulting clusters are soft clusters or hard clusters [45]. An important criterion often less studied is the way the documents are represented such as the vector space model [32], the graph model [31] and the suffix tree model [40]. [38] have performed a comprehensive comparative study of different hierarchical (divisive and agglomerative) and partitional algorithms for document clustering and have shown that the bisecting k-means algorithm outperforms other clustering techniques. D. Arotaritei and S. Mitra [2] provides a good summary of web clustering in the fuzzy framework.

Each of these classifications and algorithms has shown to be effective in the document clustering domain. No single algorithm clearly outperforms the other as different comparative studies have shown conflicting results [12, 30, 32, 38, 44]. I. Yoo and X. Hu [38] has shown that over a large and different number of datasets, bisecting k-means using the vector space model outperforms the other hierarchical and partitional clustering algorithms and the suffix tree model, but [30, 31] have shown that the graph model for representation of a document outperforms the vector space Model. [31, 30] generate hard clustering using the graph model whereas [45] has shown that soft clustering is more effective than hard clustering.

Other notable clustering algorithms used in the domain of web page clustering are [47, 40], but these algorithms do not produce hierarchical fuzzy clusters. A comprehensive survey on the topic of clustering web pages is by Carpineto et al. [10].

Our focus in this paper is information retrieval where we limit our problem to the use case that a user has one or more than one keywords and the goal is to search for documents or web pages containing those keywords within a collection of documents.

In this paper, we present a new approach to document clustering which is based on clustering keywords. These clusters are then used to regroup documents, where the degree of similarity of a document and a cluster of keywords is calculated. As a result, a document can belong to more than one cluster. Clustering words is not a new idea. [11, 33] have used word clustering to reduce dimensions of a document before eventually clustering documents. Our approach presents a solution to the information retrieval problem and has clearly different objectives than presented in [11]. Furthermore the document clusters we produce at the end are overlapping which is completely different from [11].

There are a number of Hierarchical Fuzzy Clustering algorithms proposed in the literature. G. Bordogna and Pasi [9] provides a good overview of these algorithms. A general drawback for these algorithms is the number of input parameters required to determine the number of clusters, the initial centroids and the hierarchy cut. Our approach differs from these algorithms as we do not need the user to input any parameter for execution. We have compared our results with two such algorithms that require minimal number of parameters and have shown the results in Section 6.

Many researchers [4, 6, 7, 13, 20, 23, 27] have undertaken to determine the correct parameters for clustering algorithms. Our aim is not to perform a comparative study but to show that domain dependent heuristics can help determine these parameters. In this paper, we use the co-occurrence network to achieve this task and the results shown in Section 6 reflect that the method performs well.

To compare the performance of the proposed algorithm, we use two algorithms that have similar goals to the clustering problem addressed in this paper, which is producing hierarchical fuzzy clusters for information retrieval: the Fuzzy Agglomerative Hierarchical Clustering (FAHC) Algorithm introduced by [37] and the Hierarchical -Hyperspherical Divisive Fuzzy C-Means (H2D-FCM) Algorithm [9]. The FAHC algorithm is an agglomerative fuzzy clustering algorithm which requires two parameters, the similarity threshold value and the difference threshold value. Both these values are used to determine how similar two documents should be to get clustered together. The H2D-FCM is a divisive fuzzy clustering algorithm which uses partition entropy to decide the best possible number of clusters (K) that are to be generated. Moreover the algorithm is required to run for a number of different K values before deciding the best possible value for K. For both these algorithms, we force the hierarchy of depth to three so as to avoid comparing a high depth clustering to a low depth clustering. As suggested earlier, this decision is supported by the Three-Click Rule [41]. One reason to choose these two algorithms is because one algorithm is agglomerative and the other is divisive covering two different approaches of clustering data.

2.2 Document Visualization

Different visualization systems to search and navigate through web pages have been proposed. These systems to visualize web pages can broadly be grouped into two categories: List Based Systems and Graphical Visualization systems. The list based systems keep the traditional ordered list visualization adding visual aids such as bolding words in the paragraphs [19] or clustering web pages and presenting a tree view [36, 42] along with the list. Graphical systems represent search results in a graphical environment where the visualization can either be 2D [26] or 3D [8]. The effectiveness of both list based and graphical systems has been investigated by different comparative studies but no formal proof exists and thus remains an open area of research [3]. The visualization we propose in this paper belongs to the category of Graphical Visualization Systems where we review a few systems below.

WebSearchViz [26] is a graphical system that uses the metaphor of the solar system where the user query is placed at the center and the relevant pages placed around it as a function of the similarity to the user query. It uses a vector-based similarity measure to compute the degree of relevance of a document to a keyword without clustering these documents. LightHouse [21] is an information retrieval system that integrated both the list based and graphical based visualization to represent the clusters. The visualization uses spheres to represent web pages and two spheres overlap if they are semantically very similar to each other. Although this is useful in case of a few web pages, but if many overlaps occur, it becomes difficult to visualize the web pages. [26] provides a good overview of the different visualization systems for web search results.

3 Problem Formalization

Formulating the document clustering problem, we say that given a set of n web pages represented by (p_1, p_2,…, p_n), the goal of document clustering is to group documents into K clusters such that the documents from different clusters are dissimilar from each other based on some similarity criteria. For a Fuzzy clustering algorithm, the object can belong to all of the clusters with a certain degree of membership. And for a Hierarchical clustering algorithm, each cluster K can be further divided into smaller clusters. The FAHC and the H2D-FCM algorithms use the vector space model to represent a document where each document is considered to be a vector of term frequency-inverse document frequency (tf-idf) representing keywords and given by $p_i = (w_1, w_2, \ldots ,w_m)$ where p_i refers to a document indexed 'i' and each 'w' refers to the tf-idf weight of a keyword. From this data, we construct a co-occurrence network, a graph G(V,E) where V represents the nodes or keywords (w) and E represent edges between two nodes if they appear together in a document. A node (keyword) is weighted by its frequency in the collection of documents and an edge is weighted by the number of documents in which the two keywords appear together.

These co-occurrence networks have two interesting properties. First, the node degree distribution in these networks have a long tail like structure (see Figure 4) representing the scale free degree distribution [5]. This suggests that there are nodes in the network with very high node degree or connectivity. Another important property is that every node in this network belongs to a clique. This property is inherited by construction as all the keywords extracted from a single document are connected to each other by an edge in the co-occurrence network, thus forming a clique as shown in Figure 3. We exploit both these properties to determine initial centroids and the number of clusters–the details are explained in Section V.

4 Data Sets

We have used three different data sets for experimentation. These data sets are a collection of web pages found on Wikipedia encyclopedia. These web pages were returned as a search result when Jaguar, CAC40 and Hepburn were searched as a query on the Exalead1 search engine. The web pages returned contain pages in different languages pertaining to the searched word. In each case, the top 50 results were considered and keywords from these web pages were extracted by Exalead Search Engine (http://www.exalead.com/search/wikipedia/). From each webpage, 1 to 23 keywords were extracted and the average number of keywords extracted per page for the three collections is 13. These words were chosen due to their semantic ambiguity (Jaguar), domain specificity (CAC40 Paris Stock Exchange) and affiliation to a single entity (a unique person).

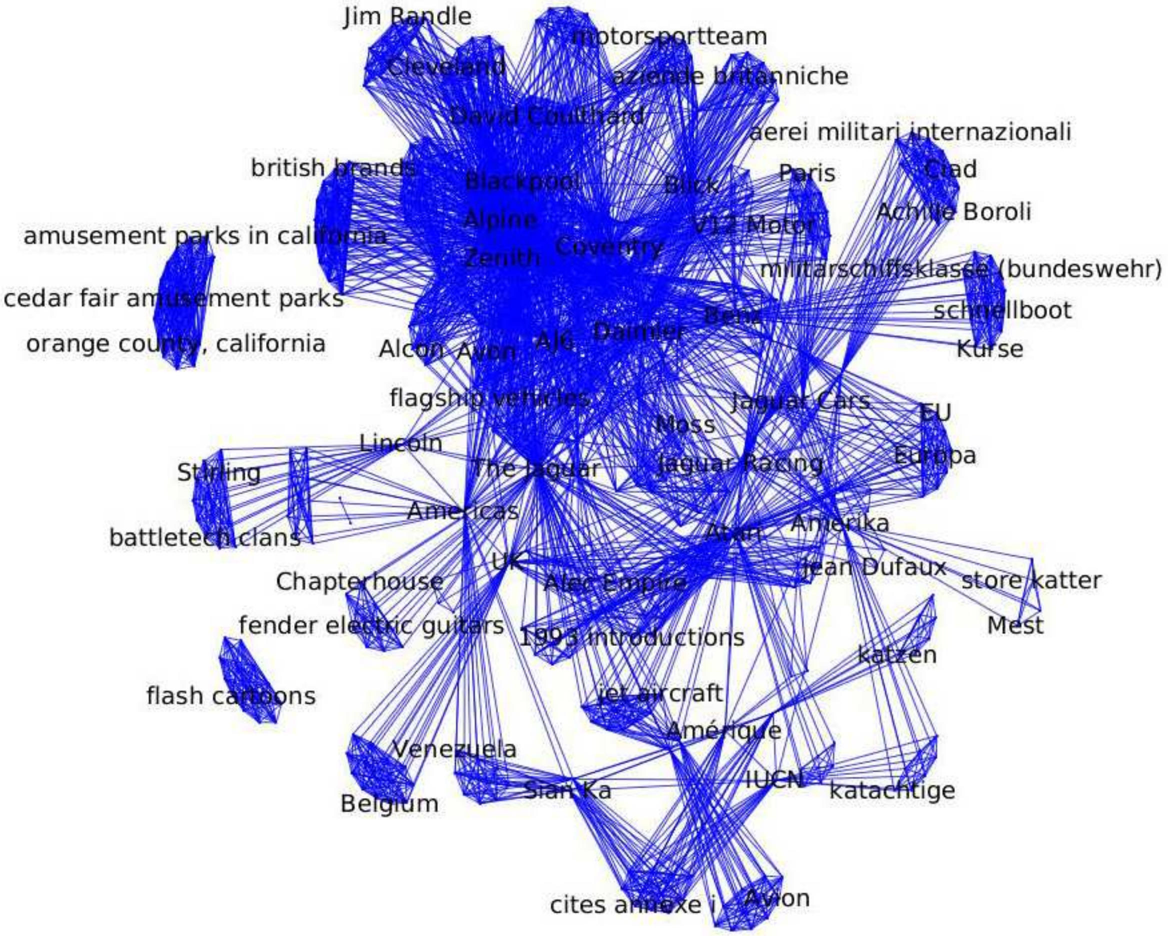


Figure 3. Co-Occurrence Network of Keyword Jaguar, Disconnected components can be easily identified as forming a clique.

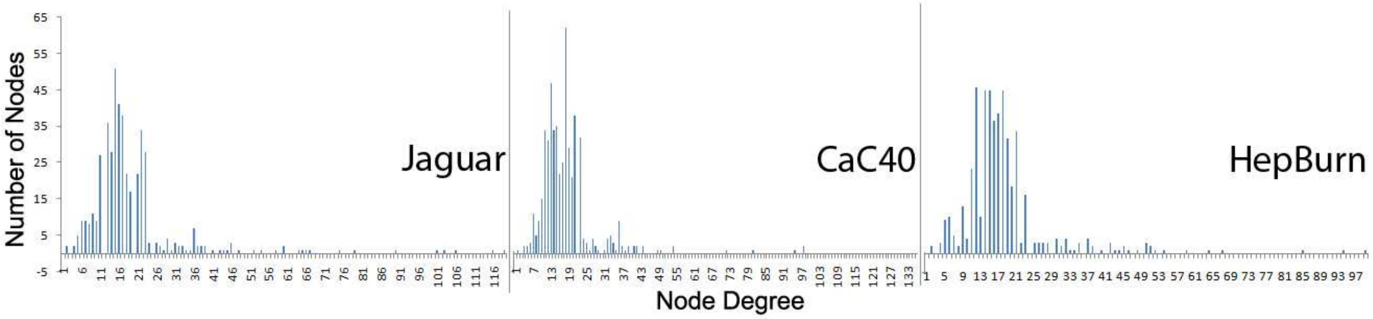


Figure 4. Degree Distribution of the three data sets.

5 Proposed Algorithm

5.1 Detection of Number of Clusters and Centroids using Topological Decomposition

In this section, we present a decomposition method to identify the important keywords in the co-occurrence network. The method exploits the fact that nodes having high degree are responsible for keeping large size networks as a single connected component. This fact can be used to identify, what we call themes or subjects around which the different web pages are organized.

The goal is to identify the keywords that have a relatively high connectivity to other nodes but are not present in all the documents. We define a Min$_d$-Degree Induced Subgraph (Min$_d$-DIS) which is an induced subgraph of nodes having degree at least *d* in graph G. Mathematically for a graph G(V, E) where V is a set of nodes and E is a set of edges, the Min$_d$-DIS is defined as G'(V', E') such that $V' \subseteq V$ and $E' \subseteq E$, $\forall u \in V'$, $Deg_G(u) \geq d$ where *d* can have values from 0 to the maximum node degree (represented by MaxDeg) possible for a network. We construct Min$_d$-DIS for d = {0,1,…,MaxDeg} to obtain a set of graphs (G_0,G_1,…,G_{MaxDeg}).

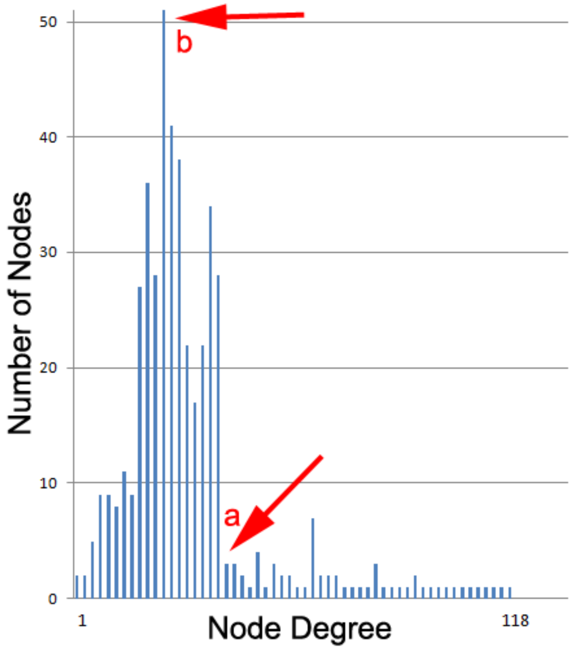


Figure 5. Degree Distribution of Jaguar Co-Occurrence Network.

Consider Figure 6(a) and (b) where the Min$_{70}$-DIS and Min$_{50}$-DIS graphs are laid out from the example Jaguar. The Min$_{70}$-DIS contains the nodes that have at least degree 70 in the co-occurrence network. All these nodes form a clique as they are connected to each other, representing that they all appear together in documents, let's call these nodes core nodes of the network. Moving from high values of node degree, we eventually come across nodes that are not connected to the core nodes in the Min$_{70}$-DIS graph.

Figure 6(b) is such an example where Min$_{50}$-DIS contains nodes that are not connected to every other node. Here we clearly see that if the core nodes (found in Min$_{70}$-DIS) are removed from this graph, we will be left with three connected components, the component with the nodes labeled Ferrari and Sports Cars, the Jaguar Cat and Atari. By repetitive application of this method, we identify these components as the basic themes of the collection of pages.

```
$Input\ G(V,E)$
$var = MaxDeg$
while $var \geq cutoff$ do
   $G' = calculate(Min_{var} - DIS)$
   if $isNotClique(nodes(G'))$ then
      $Call\ Procedure\ IdentifyThemes(G')$
   end if
   $var = var - 1$
end while
$Procedure\ IdentifyThemes(G')$
for all $i$ such that $i$ $not\ connected\ to\ all\ nodes\ in\ G'$
do
   $Find\ Nodes\ connected\ to\ i\ at\ distance\ 1\ in\ G$
   $Group\ Nodes\ as\ a\ Centroid$
end for
```

Algorithm 1. Detection of Number of Clusters and their Centroids.

We justify the use of degree of nodes by comparing it to the vector space model using tf-idf weights to represent documents. Note that the idf part of this weight assigns a low value to the keywords that are present in lots of documents. This is what we try to do when identifying themes, the keywords that are present in lots of documents can be found in what we call core nodes, thus we avoid using these nodes to determine the themes within the document collection. On the other hand, there are keywords that are present in a few documents, but are not so abundant in the entire document collection are interesting candidates to be identified as themes. Remember that the average number of keywords extracted and the maximum number of keywords extracted are not so different. We would like to mention that we tried another method called K-core decomposition [1] to extract the core keywords and themes from the collection, but the results were not semantically and empirically encouraging.

Once we have identified these basic themes, we need to find the centroids for these themes. An additional information which can be used to devise the centroids for these themes comes from the fact that by construction, in the co-occurrence network, every node belongs to a clique (as all keywords belonging to a document are connected to each other in the co-occurrence network). If we look at Figure 6(b), the word Jaguar Cat, Atari and Ferrari, Sports Cars must belong to different cliques.

Moreover, by construction of the Min$_{50}$-DIS, these nodes have a node degree of at least 50. This suggests that these nodes are important in this collection of documents. To find the collection of documents to which these themes belong, we simply group the nodes that are at distance 1 from each of these nodes in the entire network. In this way, we identify a group of documents belonging to a theme which can be used to calculate the centroid of the cluster for this theme. We also use these themes as centroid titles (see Figure 1).

Algorithm 1 requires a parameter cutoff as input which represents the value up to which the Min$_d$-DIS is calculated. To determine this value, we use a heuristic based on the degree distribution of nodes. Looking at the node degree distribution and their frequencies of the three data sets as shown in Figure 4, it is quite clear that in all these networks, there are nodes that dominate the number of connections by having a high node degree. Semantically, it is quite obvious, if we search for the word Jaguar on the web; all the pages returned will surely have this word and thus would have a very high degree as compared to the other words appearing in this collection of web pages.

To find out these high degree nodes, we calculate the slope of every two consecutive points of the degree distribution. At point 'a' in Figure 5, the slope becomes equal to zero. As the slope becomes zero or close to zero (values of - 1 or -2) the point can be considered as the cutoff point where the nodes lying after this point represents the nodes that have relatively high node degree as compared to other nodes in the network. Since our goal is to generate a hierarchical clustering, we need to generate different cutoff points to incorporate the multilevel structure. Another point that stands out in the degree distribution of these co-occurrence networks is at some value for node degree, the number of nodes attain a maximum number, as pointed by 'b' in Figure 5. Since our goal is to generate a hierarchy of up to three levels, the second cutoff is considered to be the arithmetic mean of point 'a' and 'b' whereas the third cutoff is the point 'b'.

The idea of using node degree to first decompose the entire graph, and then identify themes works because the method by construction identifies nodes that appear with a relatively high frequency in the document collection. These keywords appear in a number of documents (because of their high degree), thus they play the role of keywords that span over a number of web pages and thus, are good candidates to start identifying themes.

5.2 Hierarchical Fuzzy Clustering Algorithm based on Co-Occurrence Network (HFC-CN)

Once we have calculated the cluster centroids, the remaining algorithm to generate a fuzzy clustering is quite simple. First we associate all the remaining nodes in the co-occurrence network by assigning them to one of the centroids. Remember that the edges between any two nodes are weighted by the number of documents that appear together. This weight can be calculated for a node and a centroid by adding the weight of all the edges between a node and the nodes of a centroid. As a result of this association, we generate a hard and partitional clustering for the network. We then calculate for each document in the collection its degree or relevance to these centroids giving us a fuzzy clustering where a document can belong to more than one centroid. To generate a hierarchical clustering, we run the algorithm for different values of cutoff where at each level, only the nodes belonging to a cluster are considered rather than the whole network.

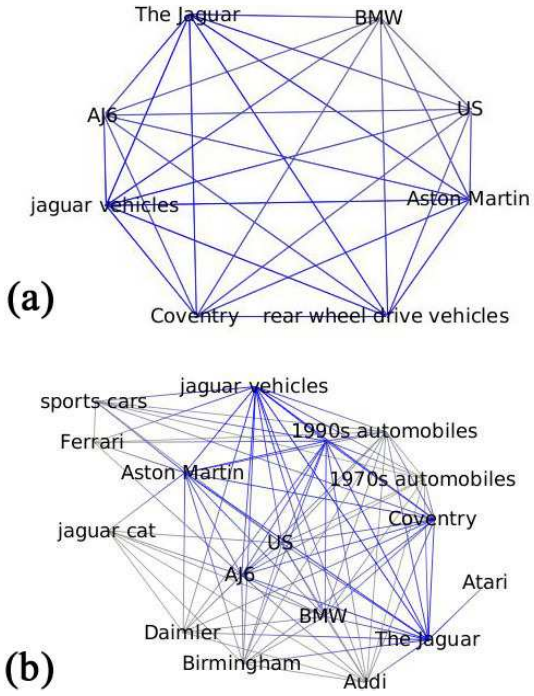
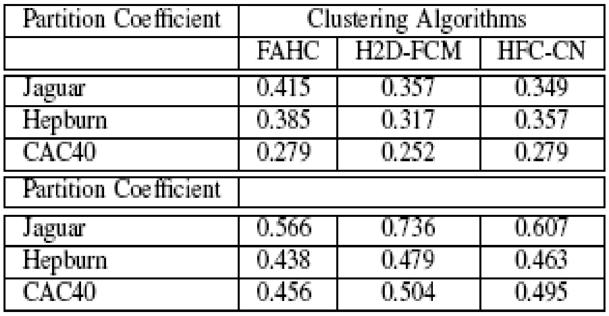


Figure 6. (a) Min70-DIS and (b) Min50-DIS for the Jaguar example

The resulting algorithm gives us a divisive hierarchical fuzzy clustering for the document collection. Each document has an associated degree of relevance to the other cluster centroid which in some cases can be zero as well. The results of the quality of clustering produced are tabulated in Table 1.

6 Experimentation and Results

To measure and compare the validity of the clustering produced by the proposed algorithm with that of FAHC and H2DFCM algorithms, we use two validity indices used in the fuzzy environment:the Partition Co-efficient (PC) [34] and Partition Efficiency (PE) [7]. Both of these methods are based on only the membership values [35] of an artifact to various clusters. The PC index indicates the average relative amount of membership sharing done between pairs of fuzzy subsets. The values range is [1/c, 1] where c is the number of clusters. The PE index is a scalar measure of the amount of fuzziness in a given fuzzy clustering where the values range is [0, log c]. In both of these cases low values indicate high clustering quality. To handle the hierarchical clustering, for each level we compute these validity indices and then we take the average over the different hierarchical levels. Note that we have forced the algorithms to produce a hierarchy of at most 3 levels.



Partition Coefficient	Clustering Algorithms		
	FAHC	H2D-FCM	HFC-CN
Jaguar	0.415	0.357	0.349
Hepburn	0.385	0.317	0.357
CAC40	0.279	0.252	0.279
Partition Coefficient			
Jaguar	0.566	0.736	0.607
Hepburn	0.438	0.479	0.463
CAC40	0.456	0.504	0.495

Table 1. Comparisons of results of clustering algorithms using partition coefficient and partition efficiency.

There are different approaches to evaluate cluster quality which can be classified as external, relative or internal. The term external validity criterion is used in the presence of ground truth [28] or when the results of the clustering algorithm can be compared with some pre-specified clustering structures [14].. Relative validity criteria measure the quality of clustering results by comparing them with the results of other clustering algorithms [22]. Internal validity criteria involve the development of functions that compute the cohesiveness of a clustering by using density, cut size, distances of entities within each cluster, or the distance between the clusters themselves etc [16, 24, 25].

For the given data sets, external validity criteria are simply not available. In the case of relative validity criteria, as Jain [18] argues, there is no clustering technique that is universally applicable in uncovering the variety of structures present in multidimensional data sets. Thus we do not have an algorithm that can generate a benchmark clustering for data sets with varying properties. For these reasons we focus on internal quality metrics only.

Table 1 shows the results obtained by the HFC-CN algorithm as compared to the other two clustering algorithms using Partition Coefficient and Partition Efficiency. These values indicate how close the elements of a cluster are to each other, and how far apart they are from elements of other clusters. Comparing values for the different algorithms, it can easily be concluded that the HFC-CN algorithm’s performance is comparable to other algorithms.

An important feature of the proposed algorithm is its different approach to calculate initial centroids. Whereas the other algorithms use only one single document as a centroid either chosen randomly, or based on the dissimilarity of existing centroids and a new document; the proposed method identifies important keywords and then calculates the initial cluster centroids based on a number of documents containing those keywords. Moreover, since the clustering is based on similarity of words as compared to similarity of documents, the topics that are similar based on some theme are grouped together and we only calculate the similarity of documents to the set of words that are clustered together. This seems to work well because semantically when we look at the clusters produced by the clustering algorithm, they are indeed coherent (see Figure 1).

Finally, we would also like to mention a few words on the efficiency of the algorithm. Other clustering algorithms based on minimizing sum-of-squares (such as the H2D-FCM algorithm) require multiple runs of the algorithm to obtain the correct value for the number of clusters K. The proposed HFC-CN algorithm uses heuristics to detect initial centroids and the number of clusters to be generated in a single pass which speeds up the clustering process.

7 Visualization

In this section, we present a visualization method for clusters, documents and keywords present in the document collection. Figure 1 represents how the clusters are visualized to give an on overview of the document collection. The Clusters are placed in a circular layout represented by round nodes and violet color. The titles of the clusters are displayed which were identified at the time of theme detection described in Section 5. All the documents are placed in an outer layer using a circular layout algorithm [46] and are represented by square shaped nodes. The documents are connected by edges to cluster centroids.

Three different colors are encoded on these documents to represent different information. The documents that belong to only a single cluster are encoded with blue color. Recall that we have a fuzzy clustering and a document can belong to more than one cluster. To avoid edge crossings and to simplify the view, we have duplicated the nodes representing a document which belongs to multiple clusters—one copy for each cluster. Although this increases the number of actual documents (outer most layer), but we are able to avoid the edge crossing problem. This technique has been used by earlier as well to reduce visual clutter and simplify drawings [17]. The nodes that are duplicated are shown in green color. To locate the presence of multiple copies of a document, or to identify how many clusters a document belongs to, we have used pink color, which is activated when a green node is selected. For example, in Figure 1, we can see that there are four instances of a document as this document belongs to four different clusters.

The edges connecting documents and clusters are encoded by a gradient from Red to Yellow representing high to low values. These colors represent the strength of membership of a document to a cluster. So, a red edge between a document and a cluster suggests that the document has high similarity as compared to edges in yellow color. A better example can be seen in Figure 7(a) where the DocumentView puts a document at the center, and the color encoding on the edges connecting to different clusters show the degree of association of the document to the clusters. From the given example, we easily see that the document titled jaguar cars is connected with high degree of association to the clusters vehicles, sports cars and jaguar but has less in common with the cluster Jacksonville Football League.

DetailView as shown in Figure 8(a) enables a user to visualize a cluster and the related web pages in an expanded view where each keyword (node) is visible to the user. In Figure 8(a), we see the cluster Atari, and some of the web pages related to this cluster. The node in the center represents the title of the web pages as named in Wikipedia. Figure 8(b) shows a portion of Figure 8(a) after zooming the bottom-right corner. The bottom-left webpage in figure 8 refers to the Atari game, bottom-right page refers to the web page on Wikipedia called jaguar homonymy disambiguity and contains the different semantic meanings related to the jaguar keyword existing in the Wikipedia database. The webpage on the right represents the webpage about the game called Alien vs. Predator based on the famous movie by James Cameron.

Note that the visualization presented in this section only supports visualization of a single level clustering. If we put all the layers in the presented layout as layers, it becomes very difficult to understand the data and focus on individual pages and keywords.

8 Conclusions and Future Research Directions

In this paper, we have presented a divisive fuzzy clustering algorithm for documents. We use graph theoretical concepts on the co-occurrence network of keywords obtained from a collection of web pages. We have addressed the well-known problems of the detection of number of clusters, the initial centroids and the depth of hierarchy to be generated in the context of information retrieval and web pages. Comparative results show that the proposed method's performance is comparable with that of two other well known Hierarchical Fuzzy Clustering algorithms. Moreover we have presented a methodology to represent the clusters, documents and their keywords to help users efficiently navigate through the collection of documents.

As part of future work, we intend to test the HFC-CN algorithm for other domains as it remains an interesting path to explore the performance of the proposed algorithm. We have not performed a formal user evaluation of the proposed visualization technique and this remains a priority to validate the efficiency of the system. Moreover, the current visualization only suits single level clustering, we tend to extend this work and find ways to enable the user to navigate and visualize a hierarchy of clustering.

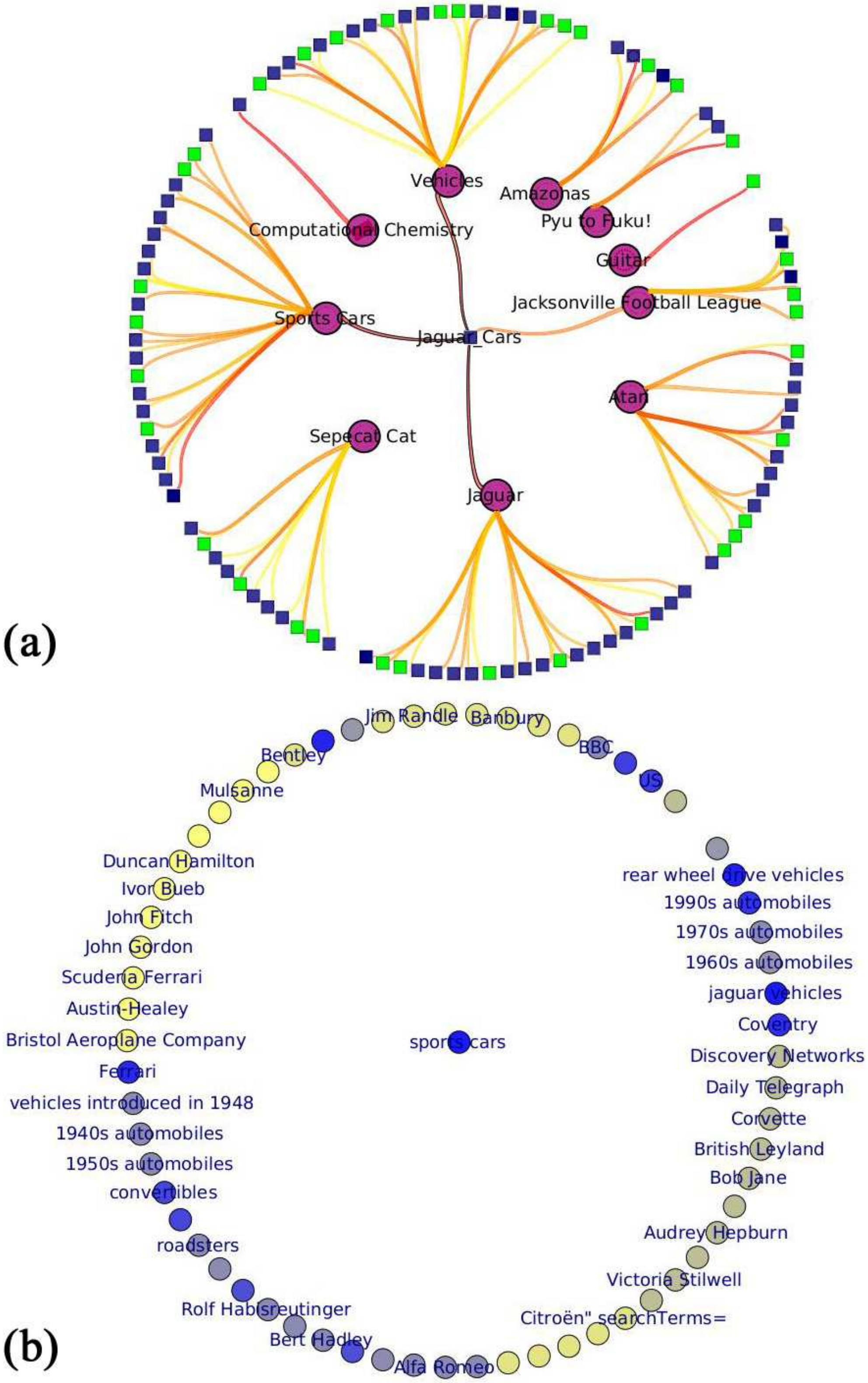


Figure 7. (a) DocumentView where a document is placed at the center and its relationship to other clusters is shown by color encoding on the edges is shown. (b) ClusterView where the centroid of a cluster can be seen, intensity from blue (high) to yellow (low) color of the represents the frequency of keywords in the entire document collection.

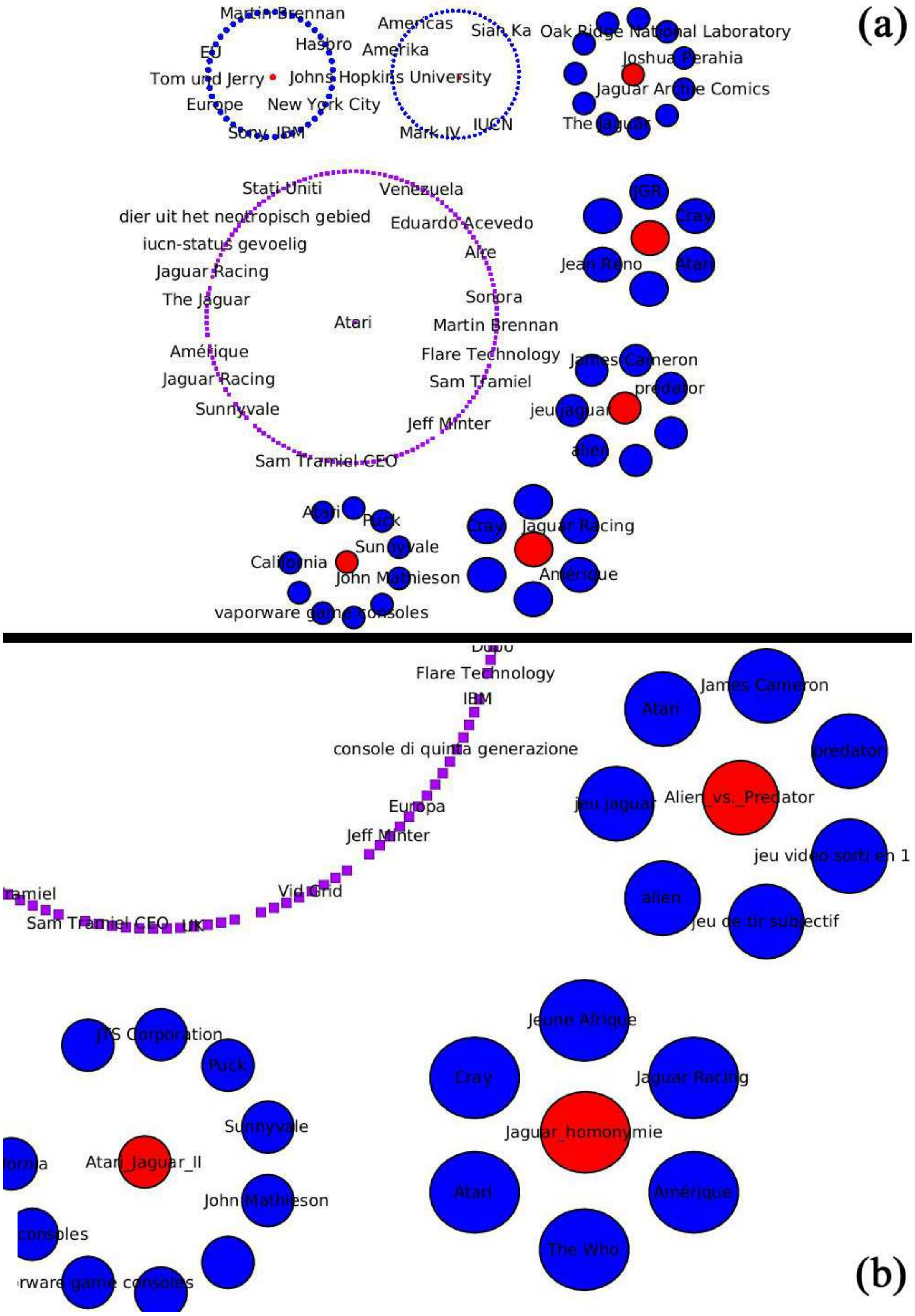


Figure 8. (a) DetailView showing the expanded view where the keywords (nodes) of a cluster and the web pages can be seen (b) A zoom-in on the right-bottom of the DetailView.

References

[1] J. I. Alvarez-Hamelin, L. Dall'Asta, A. Barrat, and A. Vespignani. k-core decomposition: a tool for the visualization of large scale networks. Advances in Neural Information Processing Systems, 18:41–50, 2006.

[2] D. Arotaritei and S. Mitra. Web mining: a survey in the fuzzy framework. Fuzzy Sets and Systems, 148(1):5 – 19, 2004. Web Mining Using Soft Computing.

[3] A. Aula. Enhancing the readability of search result summaries. In Proceedings of the HCI 2004: Design for Life, Leeds, UK., volume 2, 2004.

[4] E. Backer and A. Jain. A clustering performance measure based on fuzzy set decomposition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 3(1):66–75, 1981.

[5] A. L. Barab´asi and R. Albert. Emergence of scaling in random networks. Science, 286(5439):509–512, 1999.

[6] A. Bensaid, L. Hall, J. Bezdek, L. Clarke, M. Silbiger, J. Arrington, and R. Murtagh. Validity-guided reclustering with applications to image segmentation. IEEE Trans. Fuzzy Systems, 4(2):112–123, 1996.

[7] J. C. Bezdek. Cluster validity with fuzzy sets. Cybernetics and Systems, 3(3):58–73, 1973.

[8] N. Bonnel, V. Lemaire, A. Cotarmanac'H, and A. Morin. Effective organization and visualization of web search results. In Proceedings of the 24th IASTED International Multi-Conference Internet and Multimedia Systems and Applications, 2006.

[9] G. Bordogna and G. Pasi. Hierarchical -hyperspherical divisive fuzzy c-means (h2d-fcm) clustering for information retrieval. volume 1, pages 614–621. IEEE Computer Society, 2009.

[10] C. Carpineto, S. Osinski, G. Romano, and D. Weiss. A survey of web clustering engines. ACM Comput. Surv., 41:17:1–17:38, July 2009.

[11] I. S. Dhillon, S. Mallela, and R. Kumar. Enhanced word clustering for hierarchical text classification. In KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 191–200, 2002.

[12] S. M. Eissen, B. Stein, and M. Potthast. The suffix tree document model revisited. In Proceedings of the 5th International Conference on Knowledge Management (I-KNOW 05), pages 596–603, 2005.

[13] I. Gath and A. Geva. Unsupervised optimal fuzzy clustering. IEEE Trans. Pattern Anal. Mach. Intell., 11:773–781, 1989.

[14] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. Cluster validity methods: Part i. ACM SIGMOD Record, 31, 2002.

[15] M. Halkidi, B. Nguyen, I. Varlamis, and M. Vazirgiannis. THESUS: Organizing Web document collections based on link semantics. VLDB Journal: Very Large Data Bases, 12(4):320–332, Nov. 2003.

[16] M. Halkidi and M. Vazirgiannis. Clustering validity assessment: Finding the optimal partitioning of a data set, 2001.

[17] N. Henry, A. Bezerianos, and J.-D. Fekete. Improving the readability of clustered social networks using node duplication. IEEE Trans. Vis. Comput. Graph, 14(6):1317–1324, 2008.

[18] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. ACM Comput. Surv., 31(3):264–323, 1999.

[19] M. Kickmeier and D. Albert. The effects of scanability on information search: An online experiment. Proc. Volume 2 of the HCI 2003: Designing for Society, 2:33–36, 2003.

[20] R. Kothari and D. Pitts. On finding the number of clusters. Pattern Recognition Lett., 20:405–416, 1999.

[21] A. Leuski and J. Allan. Lighthouse: showing the way to relevant information. In InfoVis 2000. IEEE Symposium on Information Visualization, pages 125–129, 2000.

[22] O. Maimon and L. Rokach. Data Mining and Knowledge Discovery Handbook. Springer, 2005.

[23] Y. Man and I. Gath. Detection and separation of ring-shaped clusters using fuzzy clustering. IEEE Trans. Pattern Anal. Mach. Intell., 16(8):855–861, 1994.

[24] G. W. Milligan. A monte-carlo study of 30 internal criterion measures for cluster-analysis. Psychometrica, 46:187–195, 1981.

[25] Q. H. Nguyen, Rayward, and V. J. Smith. Internal quality measures for clustering in metric spaces. International Journal Business Intelligence and Data Mining, 3(1):4–29, 2008.

[26] T. Nguyen and J. Zhang. A novel visualization model for web search results. Visualization and Computer Graphics, IEEE Transactions on, 12(5):981–988, 2006.

[27] K. qi Zou, Z. ping Wang, S. jing Pei, and M. Hu5. A new initialization method for fuzzy c-means algorithm based on density. Fuzzy Information and Engineering, 54:547–553, 2009.

[28] W. M. Rand. Objective criteria for the evaluation of clustering methods. Journal of the American Statistical Association, 66(336):846–850, 1971.

[29] G. Salton and M. J. Mcgill. Introduction to Modern Information Retrieval. McGraw-Hill, Inc., 1986.

[30] A. Schenker, M. Last, H. Bunke, and A. Kandel. Classification of web documents using a graph model. Document Analysis and Recognition, International Conference on, 1:240, 2003.

[31] A. Schenker, M. Last, H. Bunke, and A. Kandel. Comparison of algorithms for web document clustering using graph representations of data. In Proc. Joint IAPR Int. Workshops SSPR and SPR, volume 3138 of LNCS, pages 190–197. Springer, 2004.

[32] M. Steinbach, G. Karypis, and V. Kumar. A comparison of document clustering techniques. Technical report, Departement of Computer Science and Engineering, University of Minnesota, 2000.

[33] W. C. Tjhi and L. H. Chen. Possibilistic fuzzy coclustering of large document collections. Pattern Recognition, 40(12):3452–3466, Dec. 2007.

[34] E. Trauwaert. On the meaning of dunn's partition coefficient for fuzzy clusters. Fuzzy Sets Syst., 25(2):217–242, 1988.

[35] W. Wang and Y. Zhang. On fuzzy cluster validity indices. Fuzzy Sets Syst., 158(19):2095–2117, 2007.

[36] Y.-f. B. Wu, L. Shankar, and X. Chen. Finding more useful information faster from web search results. In CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management, pages 568–571, 2003.

[37] Y.-C. C. C.-H. L. Yih-Jen Horng, Shyi-Ming Chen. A new method for fuzzy information retrieval based on fuzzy hierarchical clustering and fuzzy inference techniques. Fuzzy Systems, IEEE Transactions on, 13(2):216 – 228, April 2005.

[38] I. Yoo and X. Hu. A comprehensive comparison study of document clustering for a biomedical digital library medline. In JCDL '06: Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries, pages 220–229, 2006.

[39] F. Zaidi, A. Sallaberry, and G. Melancon. Revealing hidden community structures and identifying bridges in complex networks: An application to analyzing contents of web pages for browsing. In WI-IAT '09: pages 198–205, 2009.

[40] O. Zamir and O. Etzioni. Web document clustering: A feasibility demonstration. In In Proceedings of SIGIR, pages 46–54, 1998.

[41] J. Zeldman. Taking Your Talent to the Web: A Guide for the Transitioning Designer. New Riders Publishing, Thousand Oaks, CA, USA, 2001.

[42] H.-J. Zeng, Q.-C. He, Z. Chen,W.-Y. Ma, and J. Ma. Learning to cluster web search results. In SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, pages 210–217, 2004.

[43] Y. Zhang and B. Feng. A co-occurrence based hierarchical method for clustering web search results. In Web Intelligence and Intelligent Agent Technology, 2008. WI-IAT '08. IEEE/WIC/ACM International Conference on, volume 1, pages 407–410, Dec. 2008.

[44] Y. Zhao and G. Karypis. Evaluation of hierarchical clustering algorithms for document datasets. In CIKM '02: Proceedings of the eleventh international conference on Inf. and know. mgt., pages 515–524, 2002.

[45] Y. Zhao and G. Karypis. Soft clustering criterion functions for partitional document clustering: a summary of results. In D. A. Grossman, L. Gravano, C. Zhai, O. Herzog, and D. A. Evans, editors, CIKM, pages 246–247. ACM, 2004.

[46] Tollis, I. G.; Battista, G. D.; Eades, P. & Tamassia, R. Graph Drawing: Algorithms for the Visualization of Graphs Prentice Hall, 1999.

[47] Cutting, D. R.; Pedersen, J. O.; Karger, D. & Tukey, J. W. Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1992, 318-329.