Forensic Analysis Using Document Clustering And Searching

Saad S. Ansari [1] , Prof. R. B. Wagh [2]
[1, 2]Dept of Computer Engineering
[1, 2]SES's RCPIT Shirpur, MH, India

Abstract- *Now days, forensic analysis is doing crucial job in crime investigation. The crimes for investigation are embrace hacking, drug trafficking, erotica, various stealing crimes etc. In forensic analysis thousands of files are generally examined and those files contain unstructured text so it's a difficult task for forensic examiner to do such analysis in short time periods. Algorithms for clustering documents can facilitate the invention of recent and helpful data from the documents under analysis. The aim of document clustering algorithm is to grouping a set of similar documents into a cluster. Clustering algorithms are partitional K-means and Hierarchical (Single/Complete/ Average) clustering for finding relevant documents from huge amount of data and relative validity index is use to automatically estimate the number of cluster, result are show in dendrogram; results of dendrogram is helpful for expert examiner. Experts get specific documents using manual searching by keywords.*

Keywords- Forensic analysis, clustering algorithms, text mining.

I. INTRODUCTION

Forensic information provides service to governments worldwide for over one hundred years to provide correct reports of criminals. The volume of forensic information is increase exponentially. This huge quantity of data has a direct impact in computer Forensics, which might be broadly defined because the discipline that mixes components of law and technology to gather and analyze knowledge from computer systems in a way that's allowable as proof in a court of law.

Application domain typically involves examining hundreds of thousands of files per computer. This activity exceeds the expert's ability of research and interpretation of information. Therefore, methods for automated knowledge analysis, like those wide used for machine learning and data processing, are of paramount importance. Documents are in unstructured format or no structural data about textual data or there is little or no prior information about the data. Therefore clustering algorithms are generally used for preliminary data analysis[1].

The rationale behind clustering algorithms is that objects within a valid cluster are more similar to each other than they are to objects belonging to a different cluster. Thus, once a data partition has been induced from data, the expert examiner might initially focus on reviewing representative documents from the obtained set of clusters. Then, after this preliminary analysis may eventually decide to scrutinize other documents from each cluster.

Domain experts are scarce and have limited time available for performing examinations. Thus, it is reasonable to assume that, after finding a relevant document, the examiner could prioritize the analysis of other documents belonging to the cluster of interest, because it is likely that these are also relevant to the investigation. Such an approach, based on document clustering, can indeed improve the analysis of seized computers.

Clustering algorithms have been studied for decades, and the literature on the subject is huge. Therefore, we decided to choose a set of representative algorithms in order to show the potential of the proposed approach, namely: the partitional K-means [4] and hierarchical Single/ Complete/Average Link [6]. These algorithms were run with different combination of their parameters, and compare their relative performances on the studied application domain using five real world investigation cases conducted by Brazilian Federal Police Department. Use relative validity index (silhouette) to estimate the number of cluster automatically from data [1].

II. RELATED WORK

L.F.C Nassif has been proposed an approach that applies document clustering algorithms for the forensic analysis of computer devices. They illustrated an approach by carrying out wide experimentation with six well known clustering algorithms (K-mean, K-medoids, Single Link, Average Link, complete Link and CSPA) applied to five real world datasets obtained from computer seized. They were also studied uses of the comparative validity index criteria for the estimating the number of clusters in an automated manner which overcomes the limitations of previous techniques [1].

There are studies regarding use of clustering algorithms in the field of Computer Forensics and other fields related to text analysis of text documents. Most of the studies describe the use of algorithms for clustering data e.g., Expectation Maximization (EM) for unsupervised learning of Gaussian Mixture Models, K-means, Fuzzy C-means (FCM), and Self- Organizing Maps (SOM). K-means and FCM can be seen as particular cases of EM [12]. Algorithms like SOM [13]generally have inductive biases similar to K-means but are usually less computationally efficient [9].

G. Salton et al. proposed term weighting approaches in automatic text retrieval. Text indexing system based on assignment of appropriately weighted single terms produce retrieval results that are superior with other text representations. This approach summarize the insights gained in automatic term weighting provides baseline single term indexing models with which other more content analysis procedure can be compared [16].

The closest method which is relevant to our requirements is that used by Mandhani et al. in [17]. The authors combine two methodologies to evaluate their work, the first considers each cluster as a single entity and a measure is used to analyze the quality of its content (the two suggested are entropy and purity). Secondly, they analyze the resulting tree, hierarchically at each level, by looking at the number of nodes per level, the purity at each level and by comparing the generated node labels at each level. We think that this kind of hybrid analysis is the best available approach which can be applied to automatic document clustering. This approach though generates a large number of results (separate values per level). An obvious enhancement would integrate all these separate results in fewer (ideally one) values. In our work these labels are not available and hence we cannot guarantee that this mapping is always correct.

S.Oliver have proposed SOM-based algorithms were used for clustering files with intend of making the decision-making process performed by the examiners more efficient. The files were clustered by taking into account their creation dates/times and their extension. That kind of algorithm has also been used to cluster the results from keyword searches. The underlying hypothesis was that the clustered results can increase the information retrieval efficiency, because that could not be necessary to review all the documents found by the user [8].

The literature on Computer Forensics only reports the use of algorithms that assume that the number of clusters is known and fixed a priori by the user. Aimed at relaxing this assumption, which is often unrealistic in practical applications, a common approach in other domains involves estimating the number of clusters from data. Essentially, one induces different data partitions (with different numbers of clusters) and then assesses themwith a relative validity index in order to estimate the best value for the number of clusters. This work makes use of such methods, thus potentially facilitating the work of the expert examiner, who in practice would hardly know the number of clusters a priori [2], [3], [9].

III. METHODOLOGY

A. Preprocessing steps:

Before execution of clustering algorithms on text datasets,weperformed some preprocessing steps. Specially, stop words (prepositions, pronouns, articles, and irrelevant documentmetadata) are removed. Also, the Snowball stemmingalgorithm for Portuguese words has been used. Then, weadopted a conventional statistical approach for text mining, inwhich documents are described in a vector space model [14]. During this model, every document is described by a vector containing the frequencies of occurrences of words. Use a dimensionality reduction technique known as Term Variance (TV) that can increase both the effectiveness and efficiency of clustering algorithms. TV selects a number of attributes (in our case 100 words) that have the greatest variances over the documents.In order to compute distances between documents, namely: cosine-based distance [14].

B. Estimate Number of Cluster

There are many algorithms for partitioning a set of objects into k clusters, the value of k is varies from 2 to n select the best value for number of cluster is big challenge in clustering. Because each partitional algorithm like kmeans require data and number of cluster. The result of such a partitioning technique is a list of clusters with their objects, which is not as visually appealing as the dendrogram of hierarchical methods. Each cluster is represented by a silhouette, which is based on the comparison of its tightness and separation. This silhouette shows which objects lie well within their cluster, and which ones are merely somewhere in between clusters. The entire clustering is displayed by combining the silhouettes into a single plot, allowing an appreciation of the relative quality of the clusters and an overview of the data configuration [18].

In order to construct silhouettes, we only need two things: the partition we have obtained (by the application of some clustering technique) and the collection of all primitives between objects. For each object *i* we will introduce a certain value s(i).

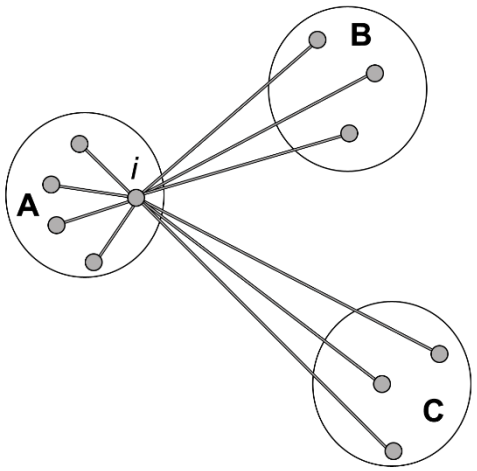

Figure 1: Computation of silhouette of object i

Let us consider an object *i* belonging to cluster *A*. The average dissimilarity of *i* to all other objects of *A* is denoted by *a(i)*. Now let us take into account cluster *C*. The average dissimilarity of object *i* to all other objects of *C* will be called *d(i,C)*. After computing *d(i,C)* for all clusters $C \neq A$, the smallest one is selected, i.e. *b(i)=min d(i,C)*, $C \neq A$ This value represents the dissimilarity of *i* to its neighbor cluster and the silhouette *s(i)* is,

$$S(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

S(i) is always lies in between -1 to 1, Thus the higher s(i) is better assignment of object *i*to a given cluster. If s(i) is equal to zero then it is not clear whether the object should have been assign to current cluster or to neighbor. Compute silhouette of all object in cluster and the find average. The best partition is has the maximum average silhouette.

C. Clustering Algorithms

The goal of clustering is to reduce the large amount of raw data by categorizing in smaller sets of similar items. In this work we used a partitioning clustering method [4] and Hierarchical clustering method [6].

a) Hierarchical Clustering Algorithm

By treating every object as a cluster and then successively merging them till we have a tendency to reach a single root cluster we have organized the data into a tree. The pair wisegrouping requires that we all know the current best (according to some criteria) clusters that either forces us to calculate the similarity each pass or do it once through memorization. The former method isn't realistic in observethus calculative this similarity matrix is needed and costs $O(n^2)$ runtime and memory [6].

Algorithm 1 Hierarchical Agglomerative Clustering

Compute the similarity matrix (Using Cosine Similarity)
Repeat
- Find two best candidates according to criterion
- Save these two in the hierarchy as sub clusters
- Insert new cluster containing elements of both clusters
- Remove the old two from the list of active clusters

Until k or one cluster remains;

Single link clustering is the cheapest and most straightforward merge criterion to use. We use the closest point in the both clusters to figure out the cluster similarity locally. In other words, the two most similar objects represent the similarity between the clusters as a whole. By sorting the values of the similarity matrix the merging phase can be done in linear time. This means that a total single link clustering runtime is only bound by the similarity matrix in $O(n^2)$ [6].

In a Complete link clustering the greedy rule instead tries to minimize the total cluster diameter. This makes the two furthest points in each cluster the interesting ones. This is a global feature that depends on the current structure and requires some extra computation in the merging phase. Running time for a complete link rule is $O(n^2 logn)$.

Average link clustering is a criterion that takes into account all similarities in each considered cluster instead of just the edges of each cluster. In other words the greedy rule tries to maximize cluster cohesion instead of diameter or local similarity. This however requires that we have the more information than just the similarity matrix because we need to calculate the mean of each cluster.

$$\frac{1}{|C_1|.|C_2|} \sum_{x \in C_1} \sum_{y \in C_2} \text{dist}(x, y)$$

Where C_1, C_2 are two separate clusters. Running time for UPGMA is $O(n^2 logn)$.

b) Kmeans Clustering Algorithm

Clustering documents using K-Means clustering algorithm and clustered resultant documents are moved for generating searching documents. The following algorithm illustrates that with the frequency of occurrence of keywords in a document the documents are clustered by means of cluster vectors and the clusters are recomputed again based on the weights assigned to the keywords.

K-Means is a simple but well known algorithm for grouping objects, clustering. Again all objects need to be represented as a set of numerical features. In addition the user has to specify the number of groups (referred to as k) he wishes to identify. Each object can be thought of as being represented by some feature vector in an n dimensional space, n being the number of all features used to describe the objects to cluster [4].

The algorithm then randomly chooses k points in that vector space, these points serve as the initial centers of the clusters. Afterwards all objects are each assigned to the center they are closest to. Usually the distance measure is chosen by the user and determined by the learning task. After that task is computed, for each cluster a new center is computed by averaging the feature vectors of all objects assigned to it. The process of assigning objects and recomputing centers is repeated until the process converges. The algorithm can be proven to converge after a finite number of iterations [4].

Algorithm 2 Basic kmeans algorithm

Initialize k centroids (Result of silhouette)
Object from vector model
Define: Seed point
Define: Number of iteration
Repeat
- For all the objects in input do
- Assign each element to its closest centroid
- End
- For all the centroids do
- Compute the mean of the assigned points. This mean now becomes the new centroid
- End
Until all centroids remains unchanged or iteration completed;

D. Document Search with keywords

A clustered data is not a labeled data to identify which group is having users required data. To perform searching string algorithm to search a document contains similar keywords on it. Suppose examiner wants to cluster a document with specific keywords. For example all ethical hacking reports contain a one similar keywords, if examiner wants to cluster all ethical hacking reports then he/she search keywords in search module and perform clustering on document short listed by search module. For search algorithm use `Sorting and Searching' techniques.

The frequency of occurrence of the keyword is calculated in each document using word count algorithm, and weight is assigned to the keyword according to the formula of term frequency and inverse document frequency [14], stored in vector model [16] sort all attribute with alphabetically for searching. Searching technique easily perform if all attribute in ascending order. Use Quick sort algorithm [19] of string array for attribute sorting. And also sort user's search keywords and searching all users' attribute in each document and display results.

IV. EXPERIMENTAL RESULTS

A. Dataset

In explicit, any kind of content that is digitally compliant may be subject to investigation. Within the datasets assessed in our study, for example, there are textual documents written in several languages (Portuguese and English). Such documents are originally created in several file formats, and a few of them are corrupted or are literally incomplete within the sense that they have been partially recovered from deleted information [10].

Five dataset of real world investigation case conducted by Brazilian federal police department obtain from author of our base paper Luis Filip da cruzNassif, the dataset is in csv file format that contain relative frequency of words per documents. Due to privacy issue detail information about the document names and their respective bags of words are not provided.

The datasets contain varying amounts of documents (N), groups (K), attributes (D), singletons (S), and number of documents per cluster (#), as reportable in below table.

Table 1: Dataset Characteristics

Dataset	N	K	D	S	# largest cluster
A	37	23	1744	12	3
B	111	49	7894	28	12
C	68	40	2699	24	8
D	74	38	5095	26	17
E	131	51	4861	31	44

N contain number of documents in particular dataset and k is a number of cluster computed by silhouette and D is attributes of file during preprocessing get bags of word or attribute of file and S is Singleton, cluster contain only one object those cluster is called as singleton. #largest cluster, after performing clustering algorithm number of cluster is found then calculate which one is largest and number of object in it.

B. Results

Perform algorithm on dataset all characteristics of dataset is shown in table 1. Perform partitioning algorithm on data with different number of cluster varies from 2 to number of document in dataset and then calculate silhouette of each partition, the maximum value of silhouette is best number of cluster for dataset. And another way to perform hierarchical clustering on data we get nested partition use those partition to calculate silhouette and then get a result of number of cluster and best partition.

Clustering algorithms is performing on all attributes (Count All) as well as perform on 100 greatest variance attributes (TV>100) and calculate which one is give a best result, the following figures show result of Average link algorithm with 100 attributes uses a similarity distances.

A desirable feature of hierarchical algorithms that make them particularly interesting for expert examiners is the summarized view of the dataset in the form of a dendrogram, which is a tree diagram that illustrates the arrangement of the clusters. The root node of the dendrogram represents the whole data set (as a single cluster formed by all objects), and each leaf node represents a particular object (as a singleton cluster). The intermediate nodes, by their turn, represent clusters merged hierarchically. The height of an intermediate node is proportional to the distance between the clusters it merges. This representation provides very informative descriptions and visualization for the potential data clustering structures [5]. For the sake of illustration, (Figure 2) to (Figure 6) shows examples of dendrograms obtained by Average Link Clustering on Hundred attributes that have greatest variance over the documents (AL100). Sub trees with low height and large width represent both cohesive and large clusters. These clusters are good candidates for a starting point inspection. Moreover, the forensic examiner can, after finding a cluster of relevant documents, inspect the cluster most similar to the one just found, because it is likely that it is also a relevant cluster. This can be done by taking advantage of the tree diagram.

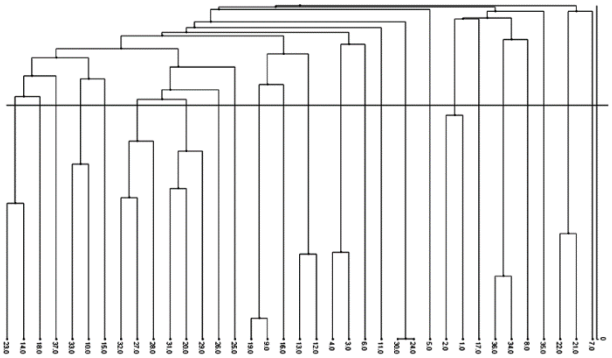

Figure 2: Dendrogram of AL100 – Dataset A

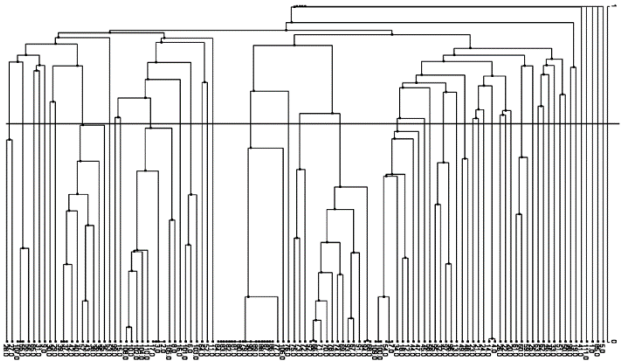

Figure 3: Dendrogram of AL100 – Dataset B

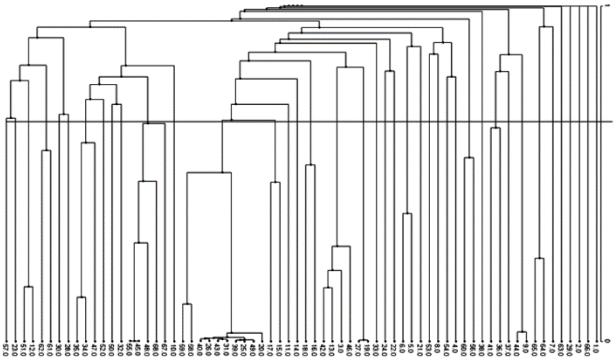

Figure 4: Dendrogram of AL100 – Dataset C

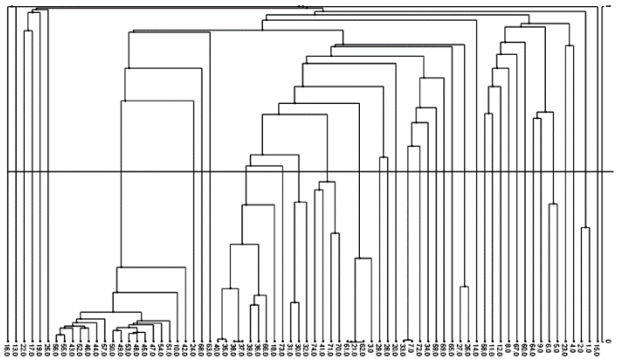

Figure 5: Dendrogram of AL100 – Dataset D

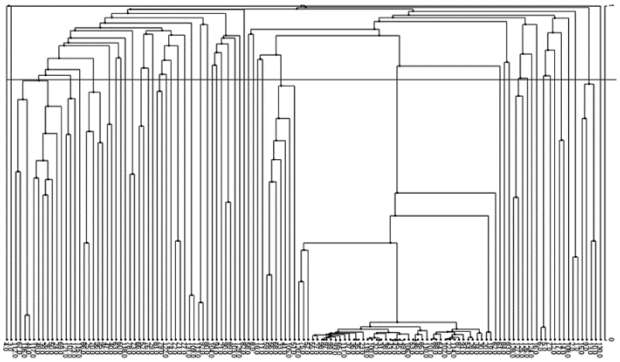

Figure 6: Dendrogram of AL100 – Dataset E

V. CONCLUSION

Clustering on large data is a very challenging problem due to the characteristics of unstructured data or no knowledge about category or class. In document clustering methods for forensic analysis of computers seized in police investigations. Also reported and discussed several practical results that can be very useful for researchers and practitioners of forensic computing. More specifically, hierarchical algorithms known as Average Link and Complete Link present a best result. The dendrograms provide offer summarized views of the documents being inspected, thus being helpful tools for forensic examiners that analyze textual documents from seized computers. As already observed in other application domains, dendrograms provide very informative descriptions and visualization capabilities of data clustering structures.

Most importantly, the clustering algorithms indeed tend to induce clusters formed by either relevant or irrelevant documents, thus contributing to enhance the expert examiner's job. Furthermore, the evaluations of approach in five real-world applications show that it has the potential to speed up the computer inspection process.

VI. ACKNOWLEDGMENT

We would like to thanks our project guide for valuable guidance and Luis Filipe da Cruz Nassif for providing a datasets.

REFERENCES

[1] L.F.D.C Nassif and E.R. Hruschka, "Document Clustering for Forensic Analysis: An Approach for Improving Computer Inspection", IEEE Transactions on Information Forensics and Security, Vol. 8, No. 1, January 2013.

[2] S. Decherchi, S. Tacconi, J. Redi, A. Leoncini, F. Sangiacomo, and R. Zunino, "Text clustering for digital forensics analysis," Computational Intelligence Security Information System, vol. 63, pp. 29–36, 2009.

[3] E. R. Hruschka, R. J. G. B. Campello, and L. N. de Castro, "Evolving clusters in gene-expression data," Information Science, vol. 176, pp. 1898–1927, 2006.

[4] A. K. Jain and R. C. Dubes, Algorithms for Clustering Data. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[5] L. Kaufman and P. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis. Hoboken, NJ: Wiley-Interscience, 1990.

[6] R. Xu and D. C.Wunsch, II, Clustering. Hoboken, NJ: Wiley/IEEE Press, 2009.

[7] R. Mundhe, A. Maind and R. Talmale, "Information Retrieval Using Document Clustering for Forensic Analysis", International Journal of Recent Advances in Engineering & Technology (IJRAET), Vol. 2, 2014.

[8] B. K. L. Fei, J. H. P. Eloff, H. S. Venter, and M. S. Oliver, "Exploring forensic data with self-organizing maps", Internatinal Conference Digital Forensics, 2005

[9] R. Hadjidj, M. Debbabi, H. Lounis, F. Iqbal, A. Szporer, and D. Benredjem, "Towards an integrated e-mail forensic analysis framework," Digital Investigation, Elsevier, vol. 5, no. 3–4, pp. 124–137, 2009.

[10] A. L. N. Fred and A. K. Jain, "Combining multiple clusterings using evidence accumulation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 6, pp. 835–850, Jun. 2005.

[11] L. Vendramin, R. J. G. B. Campello, and E. R. Hruschka, "Relative clustering validity criteria: A comparative overview," Statist. Anal. Data Mining, vol. 3, pp. 209–235, 2010.

[12] C. M. Bishop, Pattern Recognition and Machine Learning. New York: Springer-Verlag, 2006.

[13] S. Haykin, Neural Networks: A Comprehensive Foundation. Engle- wood Cliffs, NJ: Prentice-Hall, 1998.

[14] G. Salton and C. Buckley, "Term weighting approaches in automatic text retrieval," Information Process. Management, vol. 24, no. 5, pp. 513–523, 1988.

[15] G. Salton and C.S. Yang, "A Vector Space Model for Automatic Indexing",Information Retrieval and Language processing, vol. 8, page. 613-620, Nov 2011.

[16] BhushanMandhani, Sachindra Joshi, and Krishna Kummamuru. "A matrix density based algorithm to hierarchically co-cluster documents and words". In Proceedings of the twelfth international conference on World WideWeb, pages 511-518, ACM Press, 2003.

[17] Peter J. ROUSSEEUW, "Silhouettes: a graphical aid to the interpretation andvalidation of cluster analysis", Journal of Computational and Applied Mathematics, vol. 20, pp. 53-65, nov-2000.

[18] Ahmed M. Aliyu, Dr. P. B. Zirra, "A Comparative Analysis of Sorting Algorithmson Integer and Character Arrays", The International Journal of Engineering and Science, Vol-2,issue 7, pages 25-30,2013.