www.ijiarec.com

ISSN: 2348-2079

MAY-2015

International Journal of Intellectual Advancements and Research in Engineering Computations

AN APPROACH TO CLUSTER DOCUMENTS FOR IMPROVING COMPUTER INSPECTION IN DIGITAL FORENSIC ANALYSIS

[1]S.Padma Sudha, [2]S.Prema.

ABSTRACT

In computer forensic examination, hundreds and thousands of files are generally inspected, typically in the interest of figuring out what had happened, when it had happened, how it had happened, and finally who was involved in the crime. This might be done for the purpose of performing a root cause analysis of a computer system that had failed or is not operating in a proper manner as it should, or to figure out who is the primary cause for misuse of computer systems, or perhaps to find out who had committed a crime using a computer system or against a computer system. Ample of the data in those files comprises of text without formal organization or structure and therefore called as unstructured text. The analysis of these types of texts by computer examiners is a hardship to be performed. In this circumstance, automated procedures of examination are of great interest. Especially, documents clustering algorithms can render the disclosure of useful and new knowledge from the documents that are under examination or investigation. We propose an approach that will apply document clustering algorithms to forensic examination of computer systems seized in police investigations. We have illustrated the approach that is proposed by carrying out experimentation with K-ROSE (K- Rough Outlier Set Extraction) and hierarchical agglomerative approach (Single link, Complete Link, Average link) applied to datasets obtained from computers seized in investigations by police department. Experiments were performed with distinct combinations of parameters. Our experiments have shown that the Complete Link and Average Link algorithms produce the optimum results for our application realm. If suitably initialized, partitional K-Rose algorithm also yields to very good results. Lastly, we also present and discuss the modules that help investigators of forensic computing.

Keywords—clustering; forensic examination; keyword detection; document indexing; forensic datasets

INTRODUCTION

The volume, velocity and variety of data are enormously growing day by day. It is estimated that the volume of data in the digital world increased from 161 hexabytes in 2006 to 988 hexabytes in 2010which is about 18 times the amount of information present in all the books ever written and it continues to grow more and more rapidly. This enormous amount of data has a direct impact in Computer Forensics, which is widely defined as the branch or field that combines elements of law and computer science to collect and examine data from computer systems in a method that is admissible as proof or evidence in a court of law. In most cases, information that is gathered during a computer forensics investigation is not typically available or viewable by the average user of a computer system, such as deleted or removed files and broken bits of data that can be found in the space allocated for existing files - known by computer forensic practitioners as limp space. Special expertise and devices are needed to obtain this type of information or proof.

In our particular application realm, it usually involves investigating hundreds and thousands of files per computer system. This process surpasses the expert's potential of inspection and interpretation of data. Therefore, procedures for automated data examination, like those broadly used for data mining, are of supreme importance. Algorithms for clustering

Author for Correspondence:

[1]Final year ME, Department of Computer Science and Engineering, Mahendra Institute of Technology, Namakkal, Tamilnadu, India .

[2]Asst.Prof/M.E., Department of Computer Science and Engineering, Mahendra Institute of Technology , Namakkal, Tamilnadu, India.

are typically used for investigative data examination, where there is scant or no initial prior knowledge regarding the data. This is exactly the instance in numerous applicative implementations of Computer Forensics, comprising the one indicated in our method. From a technical point of view, our datasets composed of unlabeled data or objects (the set or groups of documents found are a priori unknown). Moreover, even presuming that these labeled datasets could possibly be obtainable from previous surveys, there are nearly no aspiration that the similar groups or classes would be still valid for the forthcoming data, acquired from other computer systems and related to distinct investigation processes. More exactly, it is more probably that these new data sample would arrive from a distinct population. In this circumstance, the use of clustering algorithms, which are efficient of finding dormant patterns from textual documents present in seized computer systems, can intensify the investigation performed by the forensic expert examiner.

The logic hidden behind algorithms for clustering is that objects or data within a cluster are more alike to each other than to the objects of a distinct cluster. Thus, once a data partition has been generated from data, the forensic expert may initially focus on examining representative documents from the acquired group of clusters. Then, after this introductory analysis, the forensic examiner may eventually decide to inspect other documents from all clusters. By doing so, the expert examiner can avoid the hard job of investigating all the documents separately.

In a more practical and realistic scenario, domain experts (e.g., forensic examiners) are scarce and have limited time available for performing examinations. Thus, it is fair to presume that, after locating a relevant document, the examiner could determine the order of analysis of other documents belonging to the cluster of interest, because it is possible that these are also relevant to the investigation. Such an approach, found to be based on document clustering, can certainly improve the analysis of seized computers.

It is well-known that the number of clusters is a critical parameter of many algorithms and it is usually a priori unspecified. As far as concerned, nevertheless, the automatic estimation of the number of clusters has not been investigated in the field of Computer Forensics. Literally, we could not even pinpoint one work that is reasonably in close proximity in its application domain and that outlines the use of algorithms capable of evaluating the number of clusters. The field of Forensics on computers only outlines the usage of algorithms that presume that the number of clusters is familiar and known, and that is fixed prior by the user. Focused at relaxing this presumption, which is usually unrealistic in practical applications, a regular approach in other domains involves evaluating the number of clusters from data. Fundamentally, one generates distinct data partitions (with distinct numbers of clusters) and then assesses them with a relative validity index in order to estimate the optimum value for the number of clusters. This work makes use of such practices, thus likely easing the work of the forensic expert examiner—who in practice would hardly know the number of clusters a priori.

FORENSIC INVESTIGATION APPROACH FOR DOCUMENT CLUSTERING

PREPROCESSING

Textual information extraction once all the potentially relevant digital information has been collected, an action devoted to text extraction from files belonging to significant categories is needed. At this stage, the analyst may have availability of files that are both documental and non-documental. With consideration to documental files, textual information can be found in a plain form, i.e. in raw text files. Concerning non-documental files, it is possible to extract the external existing metadata within the related entry record of the parent directories. Indeed, each one of these files has a set of textual metadata like name, path, mac times etc., that is maintained by the file system. Moreover, some types of non documental files could have internal textual metadata stored inside the file itself by software applications (author in a Microsoft word document, exif-data in images etc.). In these cases, textual information has to be extracted by developing appropriate procedures, not discussed in the following. At this point, a collection of raw text files is ready to be further processed by the text mining tool.

The extraction of all textual information is not a trivial task: as for the authors' prime knowledge, no tool that is automatically able to perform such activity currently exists. However, some software tools implement specific functions which are useful in this context.

EVALUATING THE NUMBER OF CLUSTERS FROM DATA

In order to evaluate the number of clusters, a broadly used technique consists of obtaining a set of data partitions with distinct numbers of clusters and then selecting that particular partition that provides the optimum result according to a specific quality criterion this provides an effective, automatic platform to support the analysis of digital textual evidences, which is a key issue for homeland security. Clustering algorithms can be applied to text mining to allow the automatic recognition of some sort of structure in the analyzed set of documents. In particular, clustering is designed to discover groups in the set of documents such that the documents within a group are more similar to one another than to documents of other groups. The core idea is to provide the analyst with clusters including documents semantically related, as a starting point for determining investigation paths.

K-ROSE CLUSTERING AND DOCUMENT SUMMARIZATION

The clustering algorithms adopted in our study which are the partitional K-ROSE cluster ensemble based algorithms are popular in the data mining and machine learning fields, and hence they have been used in our survey. Considering the partitional algorithms, it is widely known that both K-ROSE and hierarchical agglomerative approach are sensitive to initialization and usually converge to represent local minima as solutions. Trying to reduce these issues, we have used a nonrandom initialization in which distant objects from each other are chosen as starting prototypes the dissimilarities between the file names, and another partition achieved with K-ROSE from the vector space model. In this case, each partition can have distinct weights, which have been varied between 0 and 1 (in increments of 0.1 and keeping their sum equals to 1).

EXPECTED KEYWORD DETECTION

We evaluate a simple approach to eliminate outliers. This perspective makes recursive usage of the silhouette. Basically, when the optimum partition chosen by the silhouette has singletons (i.e., clusters formed only by a single object), these are eliminated. Then, the process of clustering is repeated over and over again until we find a partition without singletons. At the termination of the procedure, all singletons are assimilated into the resulting data partition as single clusters. This module summarizes the clustering algorithms used in our work and their main characteristics.

PREPARING CLUSTER VECTOR

For preparing the cluster vector one will need to find top 100 words from the file on which preprocessing step is done already. Now from that document or preferably we can say data or file numerical sentences such as the sentence which has numerical word in it. This means the sentence contains date or any kind of number in it. Forensic analysis will be the last step of proposed method. At last one can find accuracy of his work.

RESULT SET

The data set for forensic analysis will be distinct number of forensic files in distinct format which contains information on which data clustering is performed by applying a dissimilar algorithm. For the processes of clustering this paper makes use of multithreading technique. Finally that data set can be used for investigation by police.

EXPERIMENTAL EVALUATION

DATASETS

Sets of files or documents that are examined in computer forensic analysis have a great deal of variety. Particularly, any type of content that is found to be digitally amenable can be considered for investigation purposes. The datasets that are examined in our study may consist of textual documents that might be written in distinct languages. Such kind of documents might have been originally produced in distinct file formats. Of those documents, some of them might have been falsified and corrupted or they might be found literally imperfect which means that they are partly retrieved from eradicated or deleted data. Data sets that are investigated in this process contain documents with different extensions like “doc”, “docx”, and “odt”. Successively, these types of documents were converted into plain text format and further preprocessed. The data partitions that are produced as a result were examined by taking into consideration that we have a reference partition for each dataset.

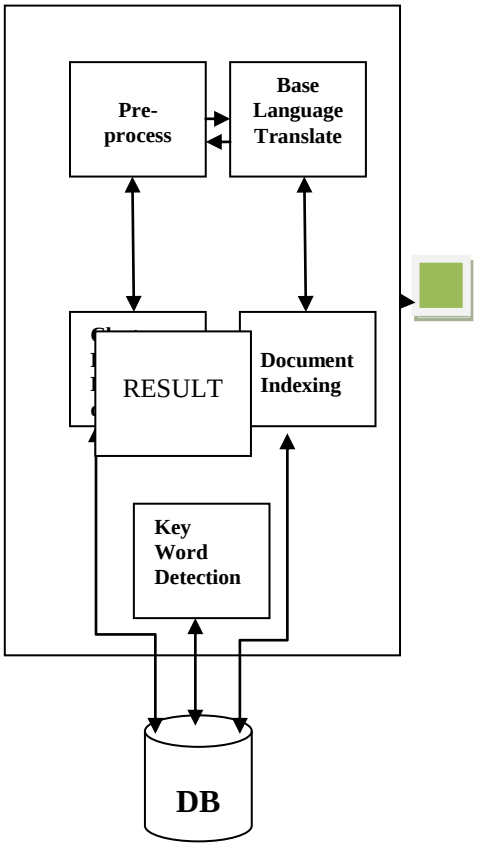

Fig. 1. K-Rose and Hierarchical Agglomerative Process

THE MEASURE OF EVALUATION

The use of reference partitions for examining data clustering algorithms is known to be a principled methodology from a scientific viewpoint. The reference partitions of our study were built by a domain specialist and usually mirrors the presumptions that the specialist has about the cluster that needs to be present in the datasets.

RESULT AND DISCUSSIONS

Generally, AL100 which is the Average Link algorithm that uses the 100 terms with the greatest variances, silhouette criterion and cosine-based similarity provided the best results with respect to both the standard deviation and the average. This provides great stability and accuracy. We also need to note that The value of ARI that is close to 1.00 signals that the specific partition is very fair or accurate with the reference partition .The values of ARI for CL100 are more likely to those that are found by AL100. The results produced by K-ROSE were also very good and more competitive to the optimum hierarchical algorithms like AL100 and the CL100.

CONCLUSION

WE PROPOSED AND presented a perspective that applies document clustering algorithms and methods to forensic analysis of computer systems that are seized in police investigations. More particularly, the hierarchical agglomerative algorithms called as Complete Link and Average Link presented the optimum results in our experiments. Focused at further enhancements, the use of algorithms for clustering data in similar applications, a promising approach for future work includes automatic investigation of approaches for the process of cluster labeling. The allotment of labels to clusters may permit the expert examiner to spot and identify the semantic content of each cluster in a quick manner even before analyzing their contents. Finally, the survey of algorithms that generates and induces overlapping partitions (e.g., Fuzzy C-Means and Expectation-Maximization for Gaussian Mixture Models) is worth of examination.

ACKNOWLEDGMENT

I would like to thank my guide and the Head of the Department of my institution and my family members, friends who rendered all possible encouragement and support during the course of this project.

REFERENCES

[1]. A. Strehl and J. Ghosh, "Cluster ensembles: A knowledge reuse framework for combining multiple partitions," *J. Mach. Learning Res.*, vol. 3, pp. 583–617, 2002.

[2]. N. L. Beebe and J. G. Clark, "Digital forensic text string searching: Improving information retrieval effectiveness by thematically clustering search results," *Digital Investigation, Elsevier*, vol. 4, no. 1, pp. 49–54, 2007.

[3]. 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.

[4]. F. Iqbal, H. Binsalleeh, B. C. M. Fung, and M. Debbabi, "Mining writeprints from anonymous e-mails for forensic investigation," *Digital Investigation, Elsevier*, vol. 7, no. 1–2, pp. 56–64, 2010.

[5]. K. Stoffel, P. Cotofrei, and D. Han, "Fuzzy methods for forensic data analysis," in *Proc. IEEE Int. Conf. Soft Computing and Pattern Recognition*, 2010, pp. 23–28.

[6]. G. Salton and C. Buckley, "Term weighting approaches in automatic text retrieval," *Inf. Process. Manage.*, vol. 24, no. 5, pp. 513–523, 1988.

[7]. 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.

[8]. Aggarwal, C. C. Charu, and C. X. Zhai, Eds., "Chapter 4: A Survey of Text Clustering Algorithms," in *Mining Text Data*. NewYork: Springer, 2012.

[9]. Y. Zhao, G. Karypis, and U. M. Fayyad, "Hierarchical clustering algorithms for document datasets," *Data Min. Knowl. Discov.*, vol. 10, no. 2, pp. 141–168, 2005.

[10]. K. Kishida, "High-speed rough clustering for very large document collections," *J. Amer. Soc. Inf. Sci.*, vol. 61, pp. 1092–1104, 2010, doi: 10.1002/asi.2131.