Text Clustering for Digital Forensics Analysis

Sergio Decherchi[1], Simone Tacconi[2], Judith Redi[1], Alessio Leoncini[1], Fabio Sangiacomo[1] and Rodolfo Zunino[1]

[1]Dept. Biophysical and Electronic Engineering, University of Genoa,
16145 Genova, Italy
{sergio.decherchi, rodolfo.zunino}@unige.it

[2]Servizio Polizia Postale e delle Comunicazioni
Ministero dell'Interno

Abstract. In the last decades digital forensics have become a prominent activity in modern investigations. Indeed, an important data source is often constituted by information contained in devices on which investigational activity is performed. Due to the complexity of this inquiring activity, the digital tools used for investigation constitute a central concern. In this paper a clustering-based text mining technique is introduced for investigational purposes. The proposed methodology is experimentally applied to the publicly available Enron dataset that well fits a plausible forensics analysis context.

Keywords: text clustering, forensics analysis, digital investigation.

1 Introduction

In the last years most of investigations performed by law enforcement agencies involve 'digital evidence', i.e. information and data of investigative value that is stored on, received, or transmitted by an digital device [1]. This evidence is acquired when data or electronic devices are seized. In this field, denominated 'digital forensics', due to increasing capability of mass storage devices, investigators have to face the problem of analysis of a great amount of information.

Key aspects of the investigational activities are the collection and analysis of available data. With this perspective digital data analysis [2-4] plays an important role in depicting a clearer vision of the context of interest.

The subsequent inquiry analysis is usually performed by a time-effort expensive human-based analysis: during this phase the analyst is requested to perform a heavy and complete study on the contents obtained from forensic acquisition.

During this activity, textual data (email, word processors etc…) constitute one of the core data sources that may contain relevant information. For this reason the typical requisite that emerges is a semi-automated texts contents analysis tool. As a consequence, in this research a two steps investigative process is proposed; these phases respectively are: text extraction and text clustering.

Text extraction is the process devoted to generate a collection of raw text file from information stored in digital devices. Text mining process relies on powerful tool to deal with large amounts of unstructured text data [5,6] possibly deriving from an investigational text extraction process.

The general area of text-mining methods comprises various approaches [6]: detection/tracking tools continuously monitor specific topics over time; document classifiers label individual files and build up models for possible subjects of interest; clustering tools process documents for detecting relevant relations among those subjects. As a result, text mining can profitably support intelligence and security activities in identifying, tracking, extracting, classifying and discovering patterns useful for building an investigative scenario.

This work addresses text clustering for forensics analysis based on a dynamic, adaptive clustering model to arrange unstructured documents into content-based homogeneous groups [7].

As benchmark the Enron emails database [8], provided the experimental domain. The research presented here shows that the document clustering framework [7] can find consistent structures suitable for investigative issues.

2 Textual data extraction

According to well known best practices of digital forensic, the first step of data extraction is the acquisition of data from devices, performed by means of a bit-stream copy, i.e. a copy of every bit of data, which includes the file slack and unallocated file space in which deleted files and e-mails are frequently recovered from.

Indeed, in the context of forensic analysis, it is common to involve deleted files, since these are often very interesting from the investigative point of view. Due to this reason, deleted file recovery constitutes the second phase of the process. For this purpose, there are two major strategies: a metadata-based and an application-based approach [9]. The first method is based on metadata of deleted files: technically speaking the related entry record of involved parent directories is recovered provided that such metadata still exist. If the metadata was reallocated to a new file or was wiped, an application based strategy is needed. In this case, chunks of data are searched for signatures that correspond to the header (start) and/or the footer (end) of known file types. This task is generally performed on the unallocated space of the file system; this stage allows also recovering files that are not pointed by any metadata, provided that their clusters were contiguously allocated. In this phase, obviously, one also extracts current files, i.e. files that are logically stored in the media.

The third phase, applied both to current files and to recovered deleted files is devoted to type identification and classification. This goal is not achievable by means of file extensions examination, since users can easily change them, but requires the analysis of headers and footers, applying to each file the same methodology of data carving.

The fourth phase is aimed to text extraction from files belonging to significant categories. In this stage, one may have both documental and non-documental files. In the case of documental files that are not purely textual, since text miner works on raw

text files, a conversion is needed. For each non-documental file, if one wants to include this kind of files, one could extract their external metadata, residing the related entry record of parent director and/or their internal metadata, stored by software application inside the file itself. At this point, a collection of raw text files is ready to be further processed by the text mining tool.

3 Text Clustering

Text mining can effectively support analysis of information sources thanks to automatic means, which is of paramount importance to homeland security [10,11].

When applied to text mining, clustering algorithms are 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 document clustering problem can be defined as follows. One should first define a set of documents $\mathcal{D} = \{D_1, \ldots, D_n\}$, a similarity measure (or distance metric), and a partitioning criterion, which is usually implemented by a cost function. In the case of flat clustering, one sets the desired number of clusters, Z, and the goal is to compute a membership function $\phi : \mathcal{D} \rightarrow \{1, \ldots, Z\}$ such that ϕ minimizes the partitioning cost with respect to the similarities among documents. Conversely, hierarchical clustering does not need to define the cardinality, Z, and applies a series of nested partitioning tasks which eventually yield a hierarchy of clusters.

3.1 Knowledge base representation

Every text mining framework should always be supported by an information extraction (IE) model [12,13] which is designed to pre-process digital text documents and to organize the information according to a given structure that can be directly interpreted by a machine learning system. Thus, a document D is eventually reduced to a sequence of terms and is represented as a vector, which lies in a space spanned by the dictionary (or vocabulary) $\mathcal{T} = \{t_j; j = 1,.., n_T\}$. The dictionary collects all terms used to represent any document D, and can be assembled empirically by gathering the terms that occurs at least once in a document collection $\mathcal{D}$; by this representation one loses the original relative ordering of terms within each document. Different models [9,10] can be used to retrieve index terms and to generate the vector that represents a document D. However, the vector space model [14] is the most widely used method for document clustering. Given a collection of documents $\mathcal{D}$, the vector space model represents each document D as a vector of real-valued weight terms $\mathbf{v} = \{w_j; j=1,..,n_T\}$. Each component of the n_T-dimensional vector is a non-negative term weight, w_j, that characterizes the j-th term and denotes the relevance of the term itself within the document D. In the following, $\mathcal{D} = \{D_u; u = 1,..,n_D\}$ will denote the corpus, holding the collection of documents to be clustered. The set $\mathcal{T} = \{t_j; j = 1,.., n_T\}$ will denote the vocabulary, which is the collection of terms that occur at least one time in $\mathcal{D}$ after the pre-processing steps of each document $D \in \mathcal{D}$ (e.g., stop-words removal, stemming [12]).

3.2 Clustering framework

The clustering strategy is mainly based on two aspects: the notion of distance between documents and the involved clustering algorithm.

According to [7] the used distance consists in a weighted Euclidean distance plus a term based on stylistic information [7]. Defined as α the weight, $\Delta^{(f)}$ the Euclidean term and $\Delta^{(b)}$ the stylistic term, then the distance between D_u and D_v can be worked out as:

$$\Delta(D_u,D_v) = \alpha \cdot \Delta^{(f)}(D_u,D_v) + (1-\alpha)\cdot \Delta^{(b)}(D_u,D_v)\ , \qquad (1)$$

Strictly speaking (1) is not a metric space because does not guarantee the triangular inequality, for this reason (1) can be more properly considered a similarity measure of data. This distance measure has been employed in the well known Kernel K-Means [7] clustering algorithm.

The conventional k-means paradigm supports an unsupervised grouping process [15], which partitions the set of samples, $\mathcal{D} = \{D_u; u= 1,..,n_D\}$, into a set of Z clusters, C_j (j = 1,…, Z). In practice, one defines a "membership vector," which indexes the partitioning of input patterns over the K clusters as: $m_u = j \Leftrightarrow D_u \in C_j$, otherwise m_u = 0; u = 1,…, n_D. It is also useful to define a "membership function" $\delta_{uj}(D_u,C_j)$, that defines the membership of the u-th document to the j-th cluster: δ_{uj} =1 if $m_u = j$, and 0 otherwise. Hence, the number of members of a cluster is expressed as

$$N_j = \sum_{u=1}^{n_D} \delta_{uj}\ ; \quad j = 1,\ldots, Z\ ; \qquad (2)$$

and the cluster centroid is given by:

$$\mathbf{w}_j = \frac{1}{N_j}\sum_{u=1}^{n_D} \mathbf{x}_u \delta_{uj}\ ; \quad j = 1,\ldots, Z\ ; \qquad (3)$$

where $\mathbf{x}_u$ is any vector-based representation of document D_u.

The kernel based version of the algorithm is based on the assumption that a function, Φ, can map any element, D, into a corresponding position, $\Phi(D)$, in a possibly infinite dimensional Hilbert space. In the new mapped space clustering centers become:

$$\Psi_j = \frac{1}{N_j}\sum_{u=1}^{n_D} \Phi_u \delta_{uj}\ ; \quad j = 1,\ldots, Z\ . \qquad (4)$$

According to [7] this data mapping allows different salient features able to ease the clustering procedure.

The ultimate result of the clustering process is the membership vector, **m**, which determines prototype positions (4) even though they cannot be stated explicitly. As per [7], for a document, D_u, the distance in the Hilbert space from the mapped image, Φ_u, to the cluster Ψ_j as per (4) can be worked out as:

$$d(\Phi_u, \Psi_j) = \left\| \Phi_u - \frac{1}{N_j} \sum_{v=1}^{n_D} \Phi_v \right\|^2 = 1 + \frac{1}{(N_j)^2} \sum_{m,v=1}^{n_D} \delta_{mj}\delta_{vj}\Phi_m \cdot \Phi_v - \frac{2}{N_j} \sum_{v=1}^{n_D} \delta_{vj}\Phi_u \cdot \Phi_v \quad . \tag{5}$$

By using expression (5), one can identify the closest prototype to the image of each input pattern, and assign sample memberships accordingly.

4 Forensic Analysis on Enron dataset

In this study, for simulating an investigational context, Enron emails dataset [8] was used. The consequence of this choice is that textual data extraction process is not explicitly performed in these experiments: this aspect does not compromise the correctness of the overall proposed approach; extracting real data from on-field devices is not an easy task due to privacy issues; for these reasons the publicly available Enron emails dataset was used.

The Enron email dataset [8] provides a reference corpus to test text-mining techniques that address investigational applications [2-4]. The Enron mail corpus was posted originally on Internet by the Federal Energy Regulatory Commission (FERC) during its investigation on the Enron case. FERC collected a total of 619,449 emails from 158 Enron employees, mainly senior managers. Each message included: the email addresses of the sender and receiver, date, time, subject, body and text. The original set suffered from document integrity problems, hence an updated version was later set up by SRI International for the CALO project [16]. Eventually, William Cohen from Carnegie Mellon University put the cleaned dataset online [8] for researchers in March 2004. Other processed versions of the Enron corpus have been made available on the web, but were not considered in the present work because the CMU version made it possible fair comparison of the obtained results with respect to established, reference corpora in the literature.

Five employees were randomly selected: *White S., Smith M., Solberg G., Ybarbo P. and Steffes J*. Collected emails (see tab.1) were separately processed, thus obtaining five different scenarios for each employee.. The underlying hypothesis was that email contents might also be characterized by the role the mailbox owner played within the company. Toward that end, when applying the clustering algorithm, only the 'body' sections of the emails were used, and sender/receiver, date/time info were discarded.

Table 1. Names and corresponding number of Emails

Name	Number of Emails
White S.	3272
Smith M.	1642
Solberg G.	1081
Ybarbo P	1291
Steffes J	3331

The performed experiments used a number of 10 clusters: this choice was guided by the practical demand of obtaining a limited number of informative groups. Tables from 2 to 7 report on the results obtained by these experiments: it shows the terms that characterize each of the clusters provided by the clustering framework for each employee. For each cluster, the most descriptive words between the twenty most frequent words of the cluster are listed; reported terms actually included peculiar abbreviations: "ect" stands for Enron Capital & Trade Resources, "hpl" stands for Houston Pipeline Company, "epmi" stands for Enron Power Marketing Inc, "mmbtu" stands for Million British Thermal Units, "dynegi" stands for Dynegy Inc, a large owner and operator of power plants and a player in the natural gas liquids and coal business, which in 2001 made an unsuccessful takeover bid for Enron.

Table 2. *Smith* results

Cluster ID	Most Frequent and Relevant Words
1	employe, business, hotel, houston, company
2	pipeline, social, database, report, link, data
3	ect, enronxg
4	coal, oil, gas, nuke, west, test, happi, business
5	yahoo, compubank, ngcorp, dynegi, night, plan
6	shank, trade
7	travel, hotel, continent, airport, flight, sheraton
8	questar, paso, price, gas
9	schedule, london, server, sun, contact, report
10	trip, weekend, plan, ski

Table 3. *Solberg* results

Cluster ID	Most Frequent and Relevant Words
1	paso, iso, empow, ub, meet
2	schedule, detected, california,iso, parsing
3	ub, employe, epe, benefit, contact, ubsq
4	shcedule, epmi, ncpa, sell, buy, peak, energi
5	dbcaps97, data, failure, database
6	trade, pwr, impact, london
7	awarded, california, iso, westdesk, portland
8	error, pasting, admin, sql, attempted
9	failure, failed, required, intervention, crawl
10	employe, price, ub, trade, energi

Table 4. *Steffes* results

Cluster ID	Most Frequent and Relevant Words
1	ferc, rto, epsa, nerc

2	market, ferc, edison, contract, credit, order, rto
3	ferc, report, approve, task, imag, attach
4	market, ee, meet, november, october
5	california, protect, attach, testimoni, whshington
6	stock, billion, financial, market, trade, investor
7	market, credit, ee, energi, util
8	attach, gov, energy, sce
9	affair, meet, report, market
10	gov, meet, november, imbal, pge, usbr

Table 5. *White* results

Cluster ID	Most Frequent and Relevant Words
1	meet, chairperson, oslo, invit, standard, smoke
2	confidential, attach, power, internet, copy
3	West, ect, meet, gas
4	gopusa, power, report, risk, inform, managment
5	webster, listserv, subscribe, htm, blank, merriam
6	report, erv, asp, efct, power, hide
7	ect, rhonda, john, david, joe, smith, michae,l mike
8	power
9	mvc, jpg, attach, meet, power, energy, canada
10	calendard, standard, monica, vacation, migration

Table 6. *YBarbo* results

Cluster ID	Most Frequent and Relevant Words
1	report, status, week, mmbtu, price, lng, lpg, capacity
2	tomdd, attach, ship, ect, master, document
3	london, power, report, impact, gas, rate, market, contact
4	dpc, transwestern, pipeline, plan
5	inmarsat, galleon, eta, telex, master, bar, fax, sea, wind
6	rate, lng, price, agreement, contract, meet
7	report, houston, dubai, dial, domest, lng, passcode
8	power, dabhol, india, dpc, mseb, govern, maharashtra
9	cargo, winter, gallon, price, eco, gas
10	arctic, cargo, methan

From the investigational point of view some interesting aspects emerges:
In *Smith* results there is an interesting cluster (cluster 10) in which the context seems not strictly adherent to the workplace usual terms. This means that analyzing that

bunch of email may mean acquiring sensible private life information potentially useful for investigation.
Solberg results do not underline any particular peculiarity. However it is curious to observe that his emails are full of server errors.
Steffes results seem extremely interesting. Cluster 6 underlines a compact cluster in which important financial aspects of Enron group are discussed. In particular some key words as F.E.R.C. (that stands for Federal Energy Regulatory Commission) are particularly expressive.
From *White* frequent terms analysis one can observe that cluster 2 contains several time the word "*confidential*" making this group interesting and worth of further analysis. Cluster 7 exhibits a strange content; it is characterized completely by names of people. This could indicate that these emails may concern private life.
YBarbo emails have no particular feature. The only aspect that can be understood is that his position is tightly linked to international affairs.

References

1. U.S. Department of Justice, Electronic Crime Scene Investigation: A Guide for First Responders, I Edition, NCJ 219941, 2008, http://www.ncjrs.gov/pdffiles1/nij/219941.pdf
2. Chen, H., Chung, W., Xu, J.J., Wang, G., Qin, Y., Chau, M.: Crime data mining: a general framework and some examples. IEEE Trans. Computer. 37, 50--56 (2004)
3. Seifert, J. W.: Data Mining and Homeland Security: An Overview. CRS Report RL31798, www.epic.org/privacy/fusion/crs-dataminingrpt.pdf (2007)
4. Mena, J.: Investigative Data Mining for Security and Criminal Detection. Butterworth-Heinemann (2003)
5. Sullivan, D.: Document warehousing and text mining. John Wiley and Sons (2001)
6. Fan, W., Wallace, L., Rich, S., Zhang, Z.: Tapping the power of text mining. Comm. of the ACM. 49, 76--82 (2006)
7. Decherchi, S., Gastaldo, P., Redi, J., Zunino, R.: Hypermetric k-means clustering for content-based document management, First Workshop on Computational Intelligence in Security for Information Systems, Genova. (2008)
8. The Enron Email Dataset, http://www-2.cs.cmu.edu/~enron/
9. Carrier, B., File System Forensic Analysis, Addison Wesley, 2005
10. Popp, R., Armour, T., Senator, T., Numrych, K.: Countering terrorism through information technology. Comm. of the ACM. 47, 36--43 (2004)
11. Zanasi, A. (eds.): Text Mining and its Applications to Intelligence, CRM and KM. 2nd edition , WIT Press (2007).
12. Manning, C. D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge (2008)
13. Baeza-Yates, R., Ribiero-Neto, B.: Modern Information Retrieval. ACM Press (1999).
14. Salton, G., Wong, A., Yang, L.S.: A vector space model for information retrieval. Journal Amer. Soc. Inform. Sci. 18, 613—620 (1975)
15. Linde, Y., Buzo, A., Gray, R.M.: An algorithm for vector quantizer design. IEEE Trans. Commun. COM-28, 84--95 (1980).
16. R. Bekkerman, A. McCallum, and G. Huang, "Automatic Categorization of Email into Folders: Benchmark Experiments on Enron and SRI Corpora." CIIR Technical Report IR-418 2004, http://www.cs.umass.edu/~ronb/papers/email.pdf