Improving Digital Forensics Through Data Mining

Chrysoula Tsochataridou Avi Arampatzis Vasilios Katos
Department of Electrical and Computer Engineering
Democritus University of Thrace
Xanthi 67100, Greece
{chrytsoc,avi,vkatos}@ee.duth.gr

Abstract— **In this paper we reflect upon the challenges a forensic analyst faces when dealing with a complex investigation and develop an approach for handling and analyzing large amounts of data. As traditional digital forensic analysis tools fail to identify hidden relationships in complex modus operandi of perpetrators, in this paper we employ data mining techniques in the digital forensics domain. We consider as a vehicle the Enron scandal, which is recognized to be the biggest audit failure in the U.S. corporate history. In particular, we focus on the textual analysis of the electronic messages sent by Enron employees, using clustering techniques. Our goal is to produce a methodology that could be applied by other researchers, who work on projects that involve email analysis. Preliminary findings show that it is possible to use clustering techniques in order to effectively identify malicious collaborative activities.**

Keywords: Digital Forensics, Email Analysis, Text Mining, Clustering, Weka, Simple K-means.

I. INTRODUCTION

The incorporation of computer technology in modern life has increased the productivity and the efficiency in several aspects of it. However, computer technology is not only used as a helpful tool that enhances traditional methodologies. In unethical hands, it can be used as a crime committing tool as well. Particularly, technically skilled criminals exploit its computing power and its accessibility to information, in order to perform, hide or aid unlawful or unethical activities. Nowadays, the number of information security incidents is increasing globally. Considering the fact that a big percentage of the total information produced is digital, arises the need of retrieving electronic evidence in a manner that doesn't affect its value and integrity.

Most of the collected digital evidence is often in the form of textual data, such as e-mails, chat logs, blogs, webpages and text documents. Due to the unstructured nature of such textual data, investigators during the stage of analysis usually employ searching tools and techniques to identify and extract useful information from a text. Obviously, the completeness of a research and the quality of an analysis relies on the experience and expertise of the investigators. Important information can be missed if a criminal intends to hide it.

In this paper, we propose an approach for handling and analyzing large amounts of textual data using a tool for data analysis and predictive modeling called Weka, which provides a collection of machine learning algorithms that perform data mining techniques. The data we experiment with are the emails of the Enron corpus. The objective is to develop a method for future investigators so that they can effectively identify and gain information from a large volume of unstructured textual data. This was accomplished first by parsing the data which were organized in folders, then by storing them into a MySQL database to better manage them and finally by performing data mining techniques to the textual data in order to draw some conclusions about the content of the emails. The proposed method is especially useful in the early stage of an investigation when the researchers may have a little clue of how to begin with.

II. RELATED WORK

In 2004, Bryan Klimt and Yiming Yang conducted email classification for the Enron dataset [1]. Their goal was to explore how to classify messages as organized by a human. In order to accomplish it, SVM (Support Vector Machine) classifier was used after they had cleaned the data from duplicate messages. Moreover, in 2005, Jitesh Shetty and Jafar Adibi created a MySQL database for the Enron dataset and statistically analyzed it [2]. In addition to this, they derived a social network from the dataset and presented a graph of it.

Concerning the text mining part, which is the extraction of knowledge from text documents, there were several tools proposed. Some of those were the Email Mining Tool (EMT) and the Malicious E-mail Tracking (MET). Those tools were developed at the Columbia University and employed data mining techniques to perform behavior analysis as well as social network analysis [3]. Furthermore, in the field of text mining, R. Al-Zaidy B. C. Fung, A. M. Youssef and F. Fortin proposed a data mining algorithm to discover and visualize criminal networks from a collection of text documents [4]. This paper also used Enron email corpus as a case study of real-life cybercrime.

III. THE ENRON CASE

The fraud investigated in this paper is the Enron scandal. The scandal was revealed in October 2001 and eventually led to the bankruptcy of the energy company Enron Corporation,

based in Houston, Texas [5]. The fall of Enron has been characterized as the greatest failure in the history of American capitalism and its collapse had a major impact on financial markets, since Enron dealt with many financial institutions and organizations. The company's collapse caused investors to lose a large amount of money and employees to lose their jobs, their medical insurances as well as their retirement funds. Additionally, it caused the dissolution of Arthur Andersen LLP, which was the audit company that performed both the internal and external accounting for Enron Corporation [6]. The federal investigations lasted 5 years and revealed the complex and illegal accounting practices that were conducted and encouraged by Enron's former executives. The investigations also came up with 31 terabytes of digital data including data from 130 computers, thousands of e-mails, and more than 10 million pages of documents, culling evidence that helped deliver convictions of the company's top executives, among others [7]. The collection of those emails is used to form our forensic methodology.

A. The Enron dataset

A few years after the scandal, a part of the digital collection was published by William Cohen, a professor at Carnegie Mellon University. The collection originally consisted of 1,500,000 emails [8]. However, some of them were withdrawn because of the complaints that former employees made, since they believed that their personal life was violated. The Enron dataset used in our project consists of 519,000 electronic messages both personal and formal, excluding the files that were attached in those mails [8]. The emails of this collection follow the RFC 5322 format and were used as a first material in order to produce a methodology that could be applied by other researchers, who work on projects that involve email analysis [9]. To achieve this goal, a certain procedure was followed.

B. Directory traversing and processing of email objects

The first step was to download the previously described dataset which included electronic messages organized in 3,500 files. Each and every one of the 151 employees that participated in this collection was represented by a unique folder. The employee's folder included other subfolders such as "inbox", "sent_items", "deleted_items" etc. Finally, inside those subfolders were the electronic messages. It is obvious that various levels of folders had to be traversed in order to reach the electronic message. This became feasible via the development of a Python script. The script conducted in the first place directory traversing and data parsing (meaning syntactic analysis) soon after the electronic message was reached.

As mentioned earlier, the messages were formulated according to the RFC 5322 format. For the purpose of data parsing, the message was separated into header and body. The header of the message included information such as the date, the electronic addresses of the sender and the recipients, the subject and the id of the message. The body part represented the text of the message. After the split of the message into those two parts, followed the syntactic analysis of the header line by line so that the necessary data could be extracted. The size of the data was large causing difficulties in storing and processing them using simple text files. Thus, the creation of a database that would solve those problems seemed reasonable.

C. Description of the MySQL Database

The data retrieved from the syntactic analysis of the message were inserted via the same Python script into the corresponding fields of the database tables. The MySQL database named "enron" consisted of 4 tables. The table "**MESSAGE**" included fields with records related to the electronic message (Date, Subject, body etc.), the table "**RECIPIENTINFO**" included fields with records that referred to the recipients of the message, the table "**OTHER_RECEIVERS**" included fields with records regarding the recipients of the BCC and CC type emails. Finally, a table named "**DELETED_MAILS**" was also created to store the data of the messages that employees used to delete. The database schema is showed in the Figure 1 below.

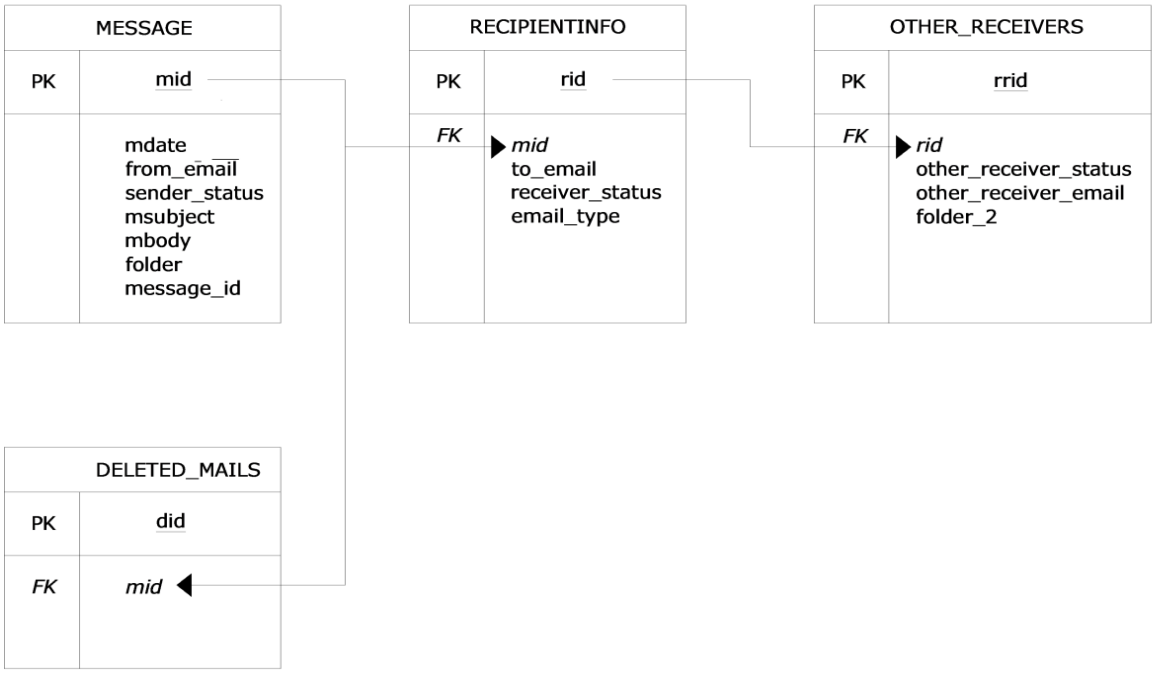

Figure 1. Database schema

IV. DATA MINING TECHNIQUES WITH WEKA

Having stored the necessary data in the MySQL database, the next step is to perform data mining techniques on them, in order to extract some useful information. For this purpose Weka is used, which is a collection of machine learning algorithms for data mining tasks [10]. Weka contains tools for data pre-processing, classification, regression, clustering, association rules and visualization [11]. This paper focuses on textual analysis of the electronic messages using clustering techniques. Therefore, a series of filters and algorithms were applied to the subject and the body of the messages. Since Weka's software is written in Java, Weka provides access to SQL databases using Java Database Connectivity (jdbc connector) and can process the result returned by a database query. In this case, msubject and mbody were loaded from the database into Weka via the execution of a query. An example of a query executed follows below.

select msubject,mbody from MESSAGE where from_email="ken.lay@enron.com";

Via the execution of the query above, the fields msubject and mbody of table "**MESSAGE"** are loaded as attributes into Weka. The messages processed and then clustered were the ones that were sent by key executives of the company. The top executives examined were Jeffrey Skilling, Kenneth Lay, Andrew Fastow, Tim Belden, Mark Koenig and Vincent Kaminski. The reason of using the "**where from_email=** ***user@enron.com*** " statement is to specify each time whose messages are going to be loaded.

A. *The StringToWordVector filter*

After the data is loaded, the filter StringToWordVector is applied to the attributes msubject and mbody. Those attributes contain instances which represent the subject and the body of the messages sent by the **x** executive. The StringtoWordVector filter pre-processes the data so that the clustering algorithm Simple K-means can later be applied. This filter converts string attributes like msubject, mbody into a set of attributes that each of them represents a single word. In other words, StringToWordVector creates a list of words which are actually attributes. These words existed inside the subject and the body of the messages. Before the application of the filter some settings have been set in the pop-up window of StringToWordVector. The settings used are the ones below:

- ***TF-IDF Transform, (term frequency–inverse document frequency)***, which is a numerical statistic used as a weighting factor that reflects how important a word is to a message of the collection. The tf-idf value increases proportionally to the number of times a word appears in a message, but is offset by the frequency of the word in the corpus [12].
- ***minTermFreq,*** which defines a minimum frequency of a word occurrence. If set on 3 for example, the

words that come up after the filter's application are those which appear at least 3 times in the collection.

- ***Stemmer,*** which removes the endings of the words. In general, stemming is the process of reducing inflected or derived words to their stem, base or root form [12].
- ***Stopwords:*** This parameter if set to "True", activates the default stopwordlist of Weka. A stopwordlist is a list of words such as *because, is, then, often* that are commonly used and do not provide important information about the content of a message.
- ***WordsToKeep:*** This parameter predetermines the number of the words that are going to appear after we apply the StringToWordVector filter.

The table below (Table I) shows the settings in the pop-up window of StringToWordVector, when the filter was applied to the string attributes msubject and mbody of Lay's sent messages. It should be noted that the settings - **minTermFrequency** and **words_to_keep** in particular - may differ between suspects, depending on the number of messages sent by each of them.

TABLE I. STRINGTOWORDVECTOR SETTINGS FOR LAY'S MESSAGES

StringToWordVector settings	Lay
IDF Transform	TRUE
TF Transform	TRUE
minTermFrequency	3
words to keep	1500
stoplist Weka 3-7-1	TRUE
stemmer	SnowballStemmer
delimeters	0123456789!#$%^&*()-=_+\/;'",.><][

B. *Simple K-means algorithm*

After the application of the filter, the string attributes msubject, mbody are converted into a list of words (dictionary), which are obviously the most frequent words that exist in the messages that sent the x executive (Kenneth Lay in the example above). The next step is to apply the Simple K-means algorithm in order to perform cluster analysis on the messages. Simple K-means is the most important algorithm for flat clustering and was chosen because of its simplicity and efficiency in performing cluster analysis in a satisfying level [12]. Cluster analysis is the task of grouping a set of messages in such way that messages from the same group (called cluster) are more similar to each other than to those in the other clusters. Simple K-means separates the emails into K groups (clusters). The K variable is an integer number and expresses the number of groups into which the messages are divided. The objective of this algorithm is the minimization of the mean squared Euclidean distance of the messages from the centers of their clusters [12]. Apart from separating the messages into K clusters, Simple K-means gives a weighting factor in the words that resulted after we applied the StringToWordVector filter. This weighting factor indicates the correlation of the word to the content of the messages that belong to a cluster. Moreover, the weighting factor depends on the word's frequency in the messages in general. The weighting factor is different for the same word from cluster to cluster. This is because one word can be more representative of the content of the messages in cluster0 than in cluster1. The biggest the factor is, the more representative of the cluster the word is. This function of Simple K-means helps in better understanding the content of clusters of messages. There is also a pop-up window for Simple K-means with a series of settings. The parameters which have been set before the application of the algorithm are:

- ***numClusters***, which determines the number of the clusters into which the messages will be separated [12].
- ***maxIterations,*** which expresses the maximum number of iterations and predetermines the implementation time of the algorithm Simple K-means [12].
- ***Seed:*** With this setting multiple iterations of the algorithm with different random initial centers can be done [12].

The first 3 rows of the table below (Table II) show the values that were given to the settings of Simple K-means pop-up window for the cluster analysis of Lay's sent messages, before the algorithm was executed. The other rows indicate Simple K-means behavior after the execution of the algorithm to the data. Similar to StringToWordVector, the settings **number_of_clusters** and **seed** may differ between the suspects examined.

TABLE II. SIMPLE K-MEANS SETTINGS FOR THE CLUSTER ANALYSIS OF LAY'S SENT MESSAGES

SimpleKmeans settings	Lay
number of clusters	5
max iterations	30
seed	8
iterations needed	2
sum of squared errors	1815,91
attributes	598

The final values given in the parameters above were chosen based on the fact that their combination minimized the sum of squared errors. The distribution of messages into clusters is being described in the next section.

V. RESULT ANALYSIS

In this section the results that came up after the application of the Simple K-means algorithm will be analyzed. As it was previously stated, the data processed were the subject (msubject) and the body (mbody) of the messages that were sent by Enron's key "players". The cluster analysis

which was conducted resulted in the configuration of five clusters of messages. The table (Table III) below shows the distribution of Lay's sent messages per cluster.

TABLE III. DISTRIBUTION OF LAY'S SENT MESSAGES INTO 5 CLUSTERS

Clustered Instances		
cluster0	7566	94%
cluster1	255	3%
cluster2	16	0%
cluster3	70	1%
cluster4	127	2%

In the table (Table III) above, it is being obvious that the majority of messages (7566 messages) that Lay sent belongs to cluster0. The second cluster (cluster1) includes 255 messages, the third (cluster2) 16 messages, the fourth (cluster3) 70 messages and finally the fifth one (cluster4) includes 127 messages.

With the help of Microsoft Excel, Weka's results are processed. Specifically, every time an executive's messages are being clustered, a table is produced. Each column of this table represents a cluster of messages and includes the most important words of the cluster. The words for each cluster are being written in a descending order, according to the weighting factor that Simple K-means gave in each word for every cluster. The bigger the factor is, the most important the word for the cluster becomes. Consequently, the most representative words of a cluster's content possess the first positions of each column.

The table below (Table IV) shows the most frequent words that exist in every cluster of Lay's messages. Our aim is to understand the content of the messages that belong to each cluster and then make some assumptions about the concerns of the executives (in this case Lay's) through time. It is worth mentioning that the table below is just a sample. The original table was very large to fit in this paper. For this reason, we chose to show the table with the first 30 most important words for each cluster of messages.

TABLE IV. THE FIRST MOST FREQUENT AND IMPORTANT WORDS IN KENNETH LAY'S CLUSTERS OF MESSAGES

cluster0	cluster1	cluster2	cluster3	cluster4
Analyst	calendars	AFL	Tuesday	Electric
Billy	Noon	AFLCIO	open	Embedded
ChairManagement	St	CounselAFLCIO	hold	Energy
ChairmanSubject	Joannie	DC	October	Gas
CommitteeAssociate	Williamson	DSilvers	forward	HNG
DepartmentHuman	Note	Damon	meeting	Natural
LayDepartment	questions	Dsilvers	Monday	Pipeline
Leading	Call	General	Directors	Transco
LeadsProgram	Place	November	Managing	allies
Lemmons	Executive	Press	bringing	arrival
Office	announced	Release	quarterly	arrived
Program	Basis	ReleaseKen	earlier	chairman
ProgramDate	quarter	Silvers	Executive	challenges
ProgramJohn	Directors	SilversAssociate	announced	changing
RepsProgram	Managing	Street	basis	commitment
Resources	bringing	aflcio	quarter	dedication
Sherriff	quarterly	apologies	PMTo	developing
SupervisorsEmbedded	Monday	inadvertantly	Lay	directors
Worlds	meeting	omitted	MessageFrom	endeavors
WorldwideFrom	October	release	work	energy
asset	forward	doc	Kenneth	environment
broadest	Earlier	press	office	era
businesses	purpose	Washington	KennethSubject	facing
campus	Committee	attached	Rosalee	felt
cc		fax	dont	gratitude
clarity		org	left	history
contributions		Friday	process	ideas

The analysis of the tables was the part of the process that involved a significant amount of uncertainty. Based on the words of each cluster of messages and on previous research that was conducted on the Enron case, there was made an attempt on drawing some conclusions regarding the content of the messages that were sent by some of the top executives of the organization. Even by a superficial inspection of the words in the first column (Table IV) represented as cluster0, we assume that Lay's messages had to do with a program called "LEADS". There is a great chance that Enron had employed graduates from that program. Lay's messages tend to inspire the superior executives so that they help those new employees to adapt easily in Enron's environment. Those assumptions are based on the presence of the words LEADS Program, Program, graduates, worldwide, guide, supervisors, direction, importance, members, philosophy (some words cannot be seen in Table IV but exist in the original full table and are worth mentioning). After some research from external public sources, it was discovered that the UC LEADS Program is one of the most prestigious fellowships awarded by the University of California system [13]. This program supports up to nine UCLA upper-division undergraduate students in the fields of science, technology, engineering, and mathematics with educational experiences that prepare them to claim positions of leadership in academia, industry, government, and public services following the completion of a doctoral degree. This confirms our initial hypothesis for the content of Lay's message, since we know the target group of employees that Enron use to have fulfills the above criteria.

We made an effort to approach the content of the messages of the other clusters in a similar manner. Initially we examined the words of each cluster of messages in order to formulate an assumption related to the content of the emails. Then we cross-validated our hypothesis via further electronic research. Most of the messages were formal and contained professional and business plans. The executives seemed to be devoted to the company and determined to make it prosper by using any means available.

VI. CONCLUSIONS

The textual analysis of each cluster of messages was a process that provided useful information to the researcher. However, there are risks in the analysis of the results since this process is cumbersome and difficult because of the unstructured nature of the text documents. It is important that every time cross-validation of the digital evidence with other resources is conducted to reveal the truth of a fact. Concerning the clustering techniques used, the Simple K-means algorithm was efficient and performed cluster analysis in a satisfying level without significant differences in the resulting clusters when the parameters in the algorithm's pop-up window were changed for the same person. The content of the messages that belonged to each cluster was obvious enough to make our assumptions and gain a deeper idea of each executive's concerns. Moreover, neither null clusters nor large squared errors were noticed after the execution of the algorithm. Finally, with respect to the matter of large datasets, it was noticed that the process was time-consuming and inaccurate.

REFERENCES

[1] B. Klimt and Y. Yang, "*The Enron Corpus: A new dataset for email classification research*", [Online]. Available: http://www.bklimt.com/papers/2004_klimt_ecml.pdf.

[2] J. Shetty and J. Adibi, "*The enron email dataset, database schema and brief statistical report*", [Online]. Available: http://foreverdata.com/1009/Enron_Dataset_Report.pdf.

[3] S. J. Stolfo, S. Hershkop, K. Wang and O. Nimeskern, "*EMT/MET: systems for modeling and detecting errant e-mails*", Proceedings of DARPA Information Survivability Conference and Exposition, 2003.

[4] R. Al-Zaidy, B. C. Fung, A. M. Youssef and F. Fortin, "*Mining criminal networks from unstructured text documents*", [Online]. Available: http://dmas.lab.mcgill.ca/fung/pub/AFYF12diin.pdf.

[5] Wikipedia, the free encyclopedia, "*Enron scandal*", [Online]. Available: http://en.wikipedia.org/wiki/Enron_scandal.

[6] W. W. Bratton, "*Enron and the dark side of shareholder value*", [Online]. Available: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=301475.

[7] THE FBI, FEDERAL BUREAU OF INVESTIGATION, "*Digital Forensics: It's a bull market*", [Online]. Available: http://www.fbi.gov/news/stories/2007/may/rcfl050707.

[8] William W. Cohen, MLD, CMU, "*Enron Email Dataset*" ,[Online]. Available: https://www.cs.cmu.edu/~enron/.

[9] E. P. Resnick, "*Internet Message Format*", [Online]. Available: http://tools.ietf.org/html/rfc5322.

[10] Machine Learning Group at the University of Waikato, *"Data Mining:Practical Machine Learning Tools and Techniques*", [Online]. Available: http://www.cs.waikato.ac.nz/~ml/weka/book.html.

[11] I. H. Witten, E. Frank, L. Trigg, M. Hall, G. Holmes and S. J. Cunningham, "*Weka: Practical machine learning tools and techniques with Java implementation,*", Department of Computer Science, University of Waikato, New Zealand, [Online]. Available: http://www.cs.waikato.ac.nz/~ml/publications/1999/99IHW-EF-LT-MH-GH-SJC-Tools-Java.pdf.

[12] C. D. Manning, P. Raghavan and H. Schutze, "*Introduction to Information Retrieval*", Cambridge University Press, 2008.

[13] Univeristy of California, "UC LEADS", [Online]. Available: http://www.ugresearchsci.ucla.edu/ucleads.htm.