Machine Learning in Computer Forensics (and the Lessons Learned from Machine Learning in Computer Security)

Davide Ariu* Giorgio Giacinto Fabio Roli
Department of Electrical and Electronic Engineering
University of Cagliari
Piazza d'Armi, 09123, Cagliari, Italy
[davide.ariu, giacinto, roli]@diee.unica.it

ABSTRACT

In this paper, we discuss the role that machine learning can play in computer forensics. We begin our analysis by considering the role that machine learning has gained in computer security applications, with the aim of aiding the computer forensics community in learning the lessons from the experience of the computer security community. Afterwards, we propose a brief literature review, with the purpose of illustrating the areas of computer forensics where machine learning techniques have been used until now. Then, we remark the technical requirements that should be meet by tools for computer security and computer forensics applications, with the goal of illustrating in which way machine learning algorithms can be of any practical help. We intend this paper to foster applications of machine learning in computer forensics, and we hope that the ideas in this paper may represent promising directions to pursue in the quest for more efficient and effective computer forensics tools.

Categories and Subject Descriptors

D.4.6 [**Security and Protection**]; I.2 [**Artificial Intelligence**]; I.5 [**Pattern Recognition**]; K.4.1 [**Public Policy Issues**]: Abuse and crime involving computers

General Terms

Algorithms, Legal Aspects, Security

Keywords

Computer Forensics, Computer Security, Machine Learning

*This research was sponsored by the RAS (Autonomous Region of Sardinia) through a grant financed with the "Sardinia PO FSE 2007-2013" funds and provided according to the L.R. 7/2007. Any opinions, findings and conclusions expressed in this material are those of the authors and do not necessarily reflect the views of the RAS.

AISec'11, October 21, 2011, Chicago, Illinois, USA.

1. INTRODUCTION

Since the second half of the nineties, the world assisted to a digital revolution that indeed changed the lifestyle of billions of people. Internet, mobile phones and a plenty of different digital devices, became part of the everyday life of all of us. In the beginning, all this stuff seemed just being something to have fun with. Nowadays, computers (of any kind), mobile phones and "the network" all represent essential tools for the professional life of millions of people. This fact obviously means that an always increasing amount of valuable information is stored in digital form: digital photos, phone books and emails are probably just the most notable examples of this phenomena.

In this information technology age, the needs of law enforcement are changing as well. Some traditional crimes, especially those concerning finance and commerce, are continuously upgraded according to the related technological advances. In a broader perspective, the analysis of computers and digital devices becomes more and more important to assess the facts in a large number of investigative cases. Computer forensic was created to address the specific and articulated needs of law enforcement to make the most of this new form of electronic evidence.

For the sake of clarity, let us point out what do we exactly mean by the terms "computer forensics". The term "**Digital forensics**" is usually related to the disciplines of analyzing digital devices for forensics purposes. Thus, it involves not only general-purpose computers but also mobile phones, game consoles or even devices such as iPods or mp3 players. In particular "**mobile forensics**" is the discipline that analyzes mobile appliances such as smart phones, or even GPS navigation systems. "**Multimedia forensics**" is the branch of digital forensics that involves the analysis of digital media (pictures, videos and audio traces). On the other side, "**Computer forensics**", in a strict sense, applies specifically to the analysis of general purpose computers, and data storage appliances or data processing devices.

At present, research in computer forensics is still at an early stage [5]: a community clearly focused on this topic does not exist, and a clear research road-map is still missing. In particular, there is not yet a clear understanding of how machine learning can help solving computer forensics problems [5]. In spite of this, we believe that there is plenty of room to improve the existing techniques, especially with the help of machine learning algorithms. In this sense, we also believe that the computer forensic community could take advantage of the experience of the computer

security community. The reason for this is that computer security and computer forensics are rooted in the same technical background.

In order to stimulate the research in this direction, in this paper we investigate the role that machine learning could play in computer forensics applications. Section 2 briefly provides an historical perspective, in order to let the reader understand the (long) way of machine learning research before being successfully applied to computer security. In section 3 we quickly review the most recent works that propose machine learning techniques for computer forensics, in order to provide a survey of the current research activities in this field. In section 4 we highlight how different the requirements are in the case of computer security and computer forensics tools, and also describe how computer forensics peculiarities can be exploited in order to apply machine learning to this discipline. Finally, we conclude in section 5.

2. HISTORICAL PERSPECTIVE

Computer security as a discipline was first studied in the early 1970s. In that period, the approach to the discipline was quite rigorous and more oriented on the development of theoretical models than on the deployment of practical applications [6, 7]. One of the cornerstones of machine learning applications in computer security is certainly represented by the work proposed by Denning [12]. Since then, a plenty of different applications of machine learning to computer security has been proposed. Doubts about the intrinsic security of machine learning algorithms have been repeatedly raised [4]. In spite of these doubts, and of the clear understanding that the security of machine learning algorithms has to be improved [3], both the scientific community and the developers of tools for enforcing computer security, now seem to be well aware of the role that can be played by machine learning techniques in the fight against cybercrime. Examples of successful applications of machine learning exist in several areas of computer security [2, 26]. In order to get to this point, about 25 years of efforts (and billion dollars) have been spent worldwide since the first known internet-wide attack in 1988 (the "Morris Worm").

At present, the computer forensics community lives in a completely different scenario. The discipline is not much younger than computer security, since computer forensics is more than 25 years old. As early as 1984, the FBI Laboratory and other law enforcement agencies began developing programs to examine computer evidence. In 1993, the FBI hosted an *International Law Enforcement Conference on Computer Evidence* that was attended by 70 representatives of various U.S. federal, state, and local law enforcement agencies and of international law enforcement agencies. Nevertheless, computer forensics received a noteworthy attention by the computer science community only in the recent years. In fact, when in the first 2000 the consequences of cyber-attacks were reported by all the newspapers and TV channels (it was the period of "Slammer" and his friends), (almost) nobody talked yet about computer forensics. Nevertheless, almost in the same period laptops, mobile phones and GPS navigation systems began to pervade the everyday life of people. Since people were becoming more and more familiar with instant messaging platforms, emails and social networks, it was quite obvious that computers will have quickly acquired relevance also in the context of forensics investigations.

Nowadays, a lot of computer forensics problems are still unaddressed, and a concrete need exists of powerful tools for forensics analysis of computers. In the next section, we will consider some of the problems that have been addressed by the research community, and we will provide a brief overview of some recent works that proposed solutions based on machine learning. In particular, we will focus on those computer forensic problems that can be clearly formulated in terms of machine learning problems.

3. LITERATURE REVIEW

In this section we review the recent literature on computer forensics with the goal of highlighting the research directions on which applications of machine learning have been proposed.

3.1 Textual documents and E-mail Forensics

Obviously, textual documents and e-mails represent a primary source of evidence during forensics analysis. According to a recent study[1], more than 3 billions of email accounts exist worldwide, the 25% being corporate email accounts. For each business account, a number of more than one hundred of emails is estimated to be sent and received every day. These numbers clearly shows that email is a primary source of communication, and thus represents a potential source of evidence that can not be neglected. While dealing with e-mails, an important task is the authorship verification and attribution. Several works have addressed this problem, by analyzing both the structure of the e-mail document (e.g. e-mail headers, number of lines and sentences, etc.) and linguistic patterns (e.g. character count, occurrences of punctuation, vocabulary "richness", etc) [11, 17]. SVM as well as clustering algorithms were employed with promising results. For instance, in [11] a precision from 84% to 100% is achieved while retrieving the e-mails of three different authors from a corpus of 156 emails.

Solutions have been also proposed for the analysis of any kind of textual document, not only e-mails. *Iqbal et. al* recently proposed a solution based on data mining techniques for the analysis of the authorship of on-line documents [18]. *Cheng et. al* propose a solution for author gender identification [9]. By experimenting with several classification algorithms (SVM, AdaBoost and Bayesian logistic regression) this work achieved promising results (the maximum accuracy is around 80%), even if the problem is far away to be definitely solved.

3.2 Network Forensics

The analysis of the network traffic can be useful in a number of computer forensics scenarios, among which a typical case can be that of a person suspected of being responsible for a cybercrime. Unfortunately, as *Wang* noticed [31], there are at least two major technical challenges in this field: (1) Forensic analysts are overwhelmed by huge volumes of low-quality evidence; (2) Cyber attacks are becoming increasingly sophisticated. Nevertheless, several works have recently addressed issues related to network forensics. *Thonnard et al.* proposed a framework for finding similar patterns in network traces [29]. *Liao et. al* propose an approach based on fuzzy theory which is able to automatically make

[1]Email Statistics Report 2011-2015 - `http://www.radicati.com`

inference from the network traffic [22]. Unfortunately, even if the achieved results look good (more than 91% of detection rate), some doubts are left on the effectiveness of the proposed approach, since the authors provide an evaluation of their system on the DARPA dataset only [24]. *Anaya et. al* proposed a technique (based on fuzzy logic and on Artificial Neural Network) to classify network flows in normal and abnormal [1]. *Wang and Daniels* proposed a graph-based approach [31]. An evidence graph is first constructed that highlights relationships among the hosts involved in an attack. Then, with a "reasoning" step the analyst is driven in the identification of the machines that had a crucial role in the context of the attack. The authors illustrate the possible applications of the algorithm by considering three different scenarios that clearly explain how the algorithm can support the forensic analyst.

Recently, solutions have been also proposed to natively include support to network forensics and monitoring in the network infrastructure [16].

3.3 Events and Data Analysis

A critical issue in computer forensics analysis is represented by the large volume of the data to be analyzed. In fact, according to recent FBI statistics, the average case size is approximately 500 GB [13]. These data can belong to different sources (e.g. network traces, memory dumps, disk images) and they are typically analyzed by using tools that operate on only a single type of digital evidence. Some recent papers proposed solutions aimed at supporting the activity of the computer forensics expert in the analysis of the data. *Fei et al.* proposed an application of Self-Organizing Maps to detect anomalies in the Internet-behavior of computer users [14]. *Khan et al.* proposed a solution based on neural networks for the construction of a time line of the relevant events [20]. The time line is created by using four different sources of information: activities of the file system, log files, registry entries (in the case of Windows machines) and also by analyzing the free blocks and the slack space.

3.4 File Fragment Classification

File fragment classification has been probably one of the most investigated problems in computer forensics. The goal is that of establishing the type of the file from which a data fragment originates without the help of the informations provided by the file system. This can be necessary for instance in the case of recovered files for which the initial header is not available anymore. The most part of the works in this area are based on the analysis of the statistical properties (e.g. byte histograms) of the distribution of the file bytes [10, 19, 21, 23]. The underground idea is that the properties of the bytes' distribution for a certain file basically depends on the originating file type. A pioneering work was that of *McDaniel et al.* [23], where the statistical models of files were created by the means of the byte histogram and the byte frequency cross correlation. In [19], the authors use both the byte distribution and the "rate of change" (the absolute value of the difference between the values of consecutive bytes). *Li et al.* propose an application of the $n-gram$ analysis to this problem [21]. They create a different centroid for each file type and calculate the Mahalanobis distance among the file and the centroids. All the considered works achieved promising results. For instance, in [21] the authors claimed a classification accuracy higher than 90%, whereas in [10] the recall was between 90 and 100% for common file types such as Acrobat PDF or JPEG. In spite of this, to the best of our knowledge, no one of the proposed approaches has found application in a real tool. In fact, as Roussev noticed [27], the promising results achieved can not be considered statistically relevant since they have been obtained on datasets that were at maximum 500MB large.

Recently, *Garfinkel et. al.* released several *forensics data sets*[2] that the scientific community can employ as a common basis for the empirical evaluation of the algorithms developed [15]. The released datasets include file corpora, disk images, cell phone dumps, and network traces.

4. COMPUTER SECURITY AND FORENSICS

The aim of this section is to highlight similarities and differences among computer security and forensics. Our goal, is to provide a clear understanding of the requirements that a computer forensic tool should be able to meet. To do this, we compare computer security and forensics by considering three different aspects: the goals pursued by the two disciplines (section 4.1); the **requirements** that should be meet by computer security and computer forensics tools (section 4.2); the **perspective** according to which **machine learning** should be applied to the two disciplines (section 4.3).

4.1 Goals

As we already mentioned in the paper, computer security and computer forensics share the same technical background. In fact, both disciplines require a clear and in-depth understanding of how the computers' world works. What it can be probably said, is that if computer security very often involves topics related to the "network' segment', computer forensics concerns are often related to issues such as disk and file analysis. This difference can be easily explained if we see the **different goals** of the two disciplines. Computer security aims to prevent something (a cyber-attack) from happening. Since the network is still the main channel for attack propagation (let's think of drive-by-downloads attacks), the analysis of the network traffic is fundamental for attack prevention and detection. This is the reason why topics related to network monitoring (e.g. botnet/fast-flux networks detection or DNS security) receive a lot of attention by the research community. On the contrary, the computer forensics analyst works with an opposite perspective. Since he is typically asked to find evidences for a crime (that can also be a cyber-crime), he obviously works *after* the crime has been committed. Thus, since the hard-drive is the place where the information managed by a computer "persists", it results obvious that issues related to the analysis of the disk (and of the information it contains) represent one of the main topics in the computer forensics research [23, 20].

4.2 Requirements

Computer security and computer forensics exhibit different sets of requirements. We analyze computer security requirements in 4.2.1 , whereas computer forensics requirements are discussed in 4.2.2.

[2] `http://digitalcorpora.org/`

4.2.1 Computer Security Requirements

With respect to a computer security tool, we identify three key requirements: it should be able to work in **real-time**, it should not generate too many **false alarms** and should be as **autonomous** as possible. The first one descends from the fact that typically the tool should be able to prevent the attack before it occurs. Anti-virus softwares, Intrusion Prevention Systems, or even Web Application Firewalls certainly represent examples of tools that are required to detect the malicious patterns in real time. This represents a particularly severe constraint, especially when large volumes of data must be analyzed (such as for instance in the case of a network-based Intrusion Prevention System). A second fundamental requirement is related to the **false alarms** rate, that indeed must be low. This always represent a crucial point in the case of anomaly-based systems since this requirement has to be meet without affecting the generalization capability of the system. Finally, it is also desirable for the tool to be as **autonomous** as possible, requiring (possibly) no intervention by the user (at least if no attack occur).

4.2.2 Computer Forensics Requirements

The requirements in the case of a computer forensics tool are totally different. In fact, it must be considered that the forensic analysis is driven by the computer forensics analyst, and then requires a considerable **human intervention**. Depending on the scenario of the investigation, the computer forensics expert has to decide where the evidence has to be searched and what is the best way to find it. For instance, it can be searched within textual files or spreadsheets if the investigation concerns financial crimes, while the search should focus on images or on the web browser history in the case of a suspect of pedophilia, or even in the system log files if the investigation concerns some cybercrime.

Once the analysis strategy has been planned, one of the biggest challenges the forensics analyst is called to face is represented by the large volume of data that typically must be analyzed. This can easily happen if the investigation requires the analysis of the activity of a network, of a server, or of the emails exchanged by the inquired person during several years. In a similar scenario, machine learning algorithms certainly represent a resource that can be exploited to facilitate the activity of the forensics analyst. In particular, we think that the research should be pushed in the direction of developing algorithms for automatic clustering or categorization of documents. For instance, we think that algorithms of text categorization can be certainly adapted to forensics purposes [28].

A second point that can be considered is that computer forensics experts typically employ several similar tools to perform the same analysis. This basically happens because not always different tools produce the same results. Thus, using several tools can help to find the evidence that one tool could have not found, or even to have confirmation if all the tools produce the same result. Nevertheless, this raises the problem of **correlating informations** from different sources. Solutions have been proposed that address this issue [8] but, at the best of our knowledge, nothing yet has been done with the support of machine learning algorithms. In this sense, the vast literature on "alert correlation" frameworks based on machine learning algorithms can certainly provide useful hints [30] to the computer forensics community.

Finally, it must be considered that computer forensics analysis are not subject to real-time constraints. In fact, it is absolutely reasonable to have tools that require even several days of computations if, at the end, the work of the analyst will result facilitated. Actually, this opens the possibility to consider also complex and heavy algorithms.

4.3 A formal comparison

In this section, we propose a different (and more *formal*) way of comparing computer security and computer forensics based on the analysis of *Mitchell* about the place of machine learning in computer science [25]. As *Mitchell* noticed, "*machine learning methods are the best methods in applications that are too complex for people to manually design the algorithm*". In our opinion, both computer security and computer forensics fall into this category. In fact, modern computers (and computer networks) are indeed complex and they will become certainly more as computer science will continue to evolve. It is certainly true that is complex for people to manually design the algorithm in many computer security and forensics applications. In addition, situations also exist where even if it would be theoretically possible, it actually is not because for instance too many variants of the patterns to be modeled exist (e.g. malware detection). We think that whereas the computer security community is completely aware of this, the computer forensics community actually is not.

With regard to the *Mitchell's* considerations, a point on which computer security and forensics are probably different, is the need "*that the software customize to its operational environment after it is fielded*". This requirement also exists in computer security (let us think to anomaly based IDS) but we are persuaded that it is definitely stronger in the case of computer forensics applications. In fact, in computer forensics the analysis can not be approached in the same way whatever the case is, since it must be tailored to each specific investigation scenario. In this sense, the **human intervention** can certainly represent the value added on which computer forensics can rely with respect to computer security, if the learning algorithm is able to incorporate the feedback provided by the analyst.

5. CONCLUSIONS

In this paper we proposed some useful guidelines for the application of machine learning to computer forensics. We first provided an historical perspective for both computer security and forensics. Then, we briefly reviewed the literature in order to illustrate in which areas of computer forensics machine learning has been recently applied. After, we discussed differences and similarities among computer security and forensics, in order to make clear what should be expected from applications of machine learning in computer forensics. Finally, we provided for a more formal comparison of the two disciplines, in order to illustrate the perspective according to which machine learning should be applied in computer forensics.

6. REFERENCES

[1] E. Anaya, M. Nakano-Miyatake, and H. Perez Meana. Network forensics with neurofuzzy techniques. In

Circuits and Systems, 2009. MWSCAS '09. 52nd IEEE International Midwest Symposium on, pages 848 –852, August 2009.

[2] D. Ariu, R. Tronci, and G. Giacinto. HMMpayl: An Intrusion Detection System Based On Hidden Markov Models. *Computers & Security*, 30(4):221 – 241, 2011.

[3] M. Barreno, P. L. Bartlett, F. J. Chi, A. D. Joseph, B. Nelson, B. I. P. Rubinstein, U. Saini, and J. D. Tygar. Open problems in the security of learning. In D. Balfanz and J. Staddon, editors, *AISec*, pages 19–26. ACM, 2008.

[4] M. Barreno, B. Nelson, R. Sears, A. D. Joseph, and J. D. Tygar. Can machine learning be secure? In F.-C. Lin, D.-T. Lee, B.-S. P. Lin, S. Shieh, and S. Jajodia, editors, *ASIACCS*, pages 16–25. ACM, 2006.

[5] N. Beebe. Digital forensic research: The good, the bad and the unaddressed. In G. Peterson and S. Shenoi, editors, *Advances in Digital Forensics V*, volume 306 of *IFIP Advances in Information and Communication Technology*, pages 17–36. Springer Boston, 2009.

[6] D. E. Bell and L. J. LaPadula. Secure computer systems: Mathematical foundations and model. Technical Report M74244 1, MITRE Corporation Bedford MA, May 1973.

[7] K. J. Biba. Integrity considerations for secure computer systems. Technical report a423930, MITRE Corporation Bedford MA, April 1977.

[8] A. Case, A. Cristina, L. Marziale, G. G. Richard, and V. Roussev. Face: Automated digital evidence discovery and correlation. *Digital Investigation*, 5(Supplement 1):S65 – S75, 2008. The Proceedings of the Eighth Annual DFRWS Conference.

[9] N. Cheng, R. Chandramouli, and K. Subbalakshmi. Author gender identification from text. *Digital Investigation*, 8(1):78 – 88, 2011.

[10] O. de Vel. File classification using byte sub-stream kernels. *Digital Investigation*, 1(2):150 – 157, 2004.

[11] O. de Vel, A. Anderson, M. Corney, and G. Mohay. Mining e-mail content for author identification forensics. *ACM SIGMOD Record*, 30:55–64, December 2001.

[12] D. Denning. An intrusion-detection model. *Software Engineering, IEEE Transactions on*, SE-13(2):222 – 232, February 1987.

[13] FBI. RCFL Program Annual Report for Fiscal Year 2010.

[14] B. Fei, J. Eloff, H. Venter, and M. Olivier. Exploring forensic data with self-organizing maps. In M. Pollitt and S. Shenoi, editors, *Advances in Digital Forensics*, volume 194 of *IFIP International Federation for Information Processing*, pages 113–123. Springer Boston, 2005.

[15] S. Garfinkel, P. Farrell, V. Roussev, and G. Dinolt. Bringing science to digital forensics with standardized forensic corpora. *Digital Investigation*, 6:S2 – S11, 2009.

[16] P. Giura and N. Memon. Netstore: An efficient storage infrastructure for network forensics and monitoring. In S. Jha, R. Sommer, and C. Kreibich, editors, *RAID*, volume 6307 of *Lecture Notes in Computer Science*, pages 277–296. Springer, 2010.

[17] F. Iqbal, H. Binsalleeh, B. C. Fung, and M. Debbabi. Mining writeprints from anonymous e-mails for forensic investigation. *Digital Investigation*, 7(1-2):56 – 64, 2010.

[18] F. Iqbal, H. Binsalleeh, B. C. Fung, and M. Debbabi. A unified data mining solution for authorship analysis in anonymous textual communications. *Information Sciences*, In Press, Corrected Proof:–, 2011.

[19] M. Karresand and N. Shahmehri. File type identification of data fragments by their binary structure. In *Information Assurance Workshop, 2006 IEEE*, pages 140 –147, June 2006.

[20] M. Khan, C. Chatwin, and R. Young. A framework for post-event timeline reconstruction using neural networks. *Digital Investigation*, 4(3-4):146 – 157, 2007.

[21] W.-J. Li, K. Wang, S. Stolfo, and B. Herzog. Fileprints: identifying file types by n-gram analysis. In *Information Assurance Workshop, 2005. IAW '05. Proceedings from the 6th Annual IEEE SMC*, pages 64 – 71, June 2005.

[22] N. Liao, S. Tian, and T. Wang. Network forensics based on fuzzy logic and expert system. *Computer Communications*, 32(17):1881 – 1892, 2009.

[23] M. McDaniel and M. Heydari. Content based file type detection algorithms. In *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*, January 2003.

[24] J. McHugh. Testing Intrusion detection systems: a critique of the 1998 and 1999 DARPA intrusion detection system evaluations as performed by Lincoln Laboratory. *ACM Transactions on Information and System Security*, 3:262–294, November 2000.

[25] T. M. Mitchell. The discipline of machine learning. Technical Report CMU-ML-06-108, Machine Learning Department, School of Computer Science, Carnegie Mellon University, 2006.

[26] R. Perdisci, W. Lee, and N. Feamster. Behavioral Clustering of HTTP-Based Malware and Signature Generation Using Malicious Network Traces. In *NSDI*, pages 391–404. USENIX Association, 2010.

[27] V. Roussev and S. Garfinkel. File fragment classification-the case for specialized approaches. In *Systematic Approaches to Digital Forensic Engineering, 2009. SADFE '09. 4th International IEEE Workshop on*, pages 3 –14, May 2009.

[28] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34:1–47, 2002.

[29] O. Thonnard and M. Dacier. A framework for attack patterns' discovery in honeynet data. *Digital Investigation*, 5(Supplement 1):S128 – S139, 2008. The Proceedings of the Eighth Annual DFRWS Conference.

[30] A. Valdes and K. Skinner. Probabilistic alert correlation. In W. Lee, L. Mé, and A. Wespi, editors, *Recent Advances in Intrusion Detection*, volume 2212 of *Lecture Notes in Computer Science*, pages 54–68. Springer, 2001.

[31] W. Wang and T. E. Daniels. A graph based approach toward network forensics analysis. *ACM Transactions on Information and System Security*, 12:4:1–4:33, October 2008.