

# **MACHINE LEARNING BASED ADAPTIVE FORENSIC ARTIFACTS COLLECTION FROM WINDOWS REGISTRY**



## **Author**

Farrukh Shabbir

274433-MS(IS)-11-2018F

## **Supervisor**

Dr. Omar Arif

A thesis submitted in partial fulfilment of the requirements for the degree  
of Masters in Information Security (MS IS)

Department of Computing (DoC)

School of Electrical Engineering and Computer Science (SEECS)

National University of Science and Technology (NUST)

Islamabad, Pakistan

(January 2020)

## CERTIFICATE OF ORIGINALITY

I hereby declare that this research titled “**Machine Learning based Adaptive Forensic Artifacts Collection from Windows Registry**” is my own work and to the best of my knowledge. It contains material not published previously or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at NUST SEECS or at any other educational institute, except where due acknowledgment, is made in the thesis. Any contribution made to the research by others, with whom I have worked at NUST SEECS or elsewhere, is explicitly acknowledged in the thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project’s design and conception or in style, presentation and linguistic is acknowledged.

Farrukh Shabbir

274433-MS(IS)-11-2018F

## **THESIS ACCEPTANCE CERTIFICATE**

Certified that final copy of MS thesis written by Mr. **Farrukh Shabbir**, (Registration No **274433**), of **SEECs** (School/College/Institute) has been vetted by undersigned, found complete in all respects as per NUST Statutes/Regulations, is free of plagiarism, errors and mistakes and is accepted as partial fulfillment for award of MS degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have also been incorporated in the said thesis.

Signature: \_\_\_\_\_

Name of Supervisor: **Dr. Omar Arif**

Date: \_\_\_\_\_

Signature (HOD): \_\_\_\_\_

Date: \_\_\_\_\_

Signature (Dean/Principal): \_\_\_\_\_

Date: \_\_\_\_\_

## APPROVAL

It is certified that the contents and form of the thesis titled “**Machine Learning Based Adaptive Forensic Artifacts Collection from Windows Registry**” submitted by **Farrukh Shabbir** have been found satisfactory for the requirement of the degree.

Advisor: **Dr. Omar Arif**  
Signature: \_\_\_\_\_  
Date: \_\_\_\_\_

Committee Member 1: **Dr. Shahzad Saleem**  
Signature: \_\_\_\_\_  
Date: \_\_\_\_\_

Committee Member 2: **Dr. Sharifullah Khan**  
Signature: \_\_\_\_\_  
Date: \_\_\_\_\_

Committee Member 3: **Dr. Hasan Tahir**  
Signature: \_\_\_\_\_  
Date: \_\_\_\_\_

## **ACKNOWLEDGEMENTS**

I am thankful to Allah Subhana wa Taalah for being with me and nothing is possible without the will of Allah. I could not have achieved anything without Allah's ultimate support and blessings. I would like to thank my father and mother for much needed prayers and guidance throughout my life. They are such an inspiration, may Allah SWT bless them ever, Aameen. Thank you for being my parents. You have not only taught me academic subjects but the true meaning of being a Momin and a good human being. I also want to thank my brothers and sisters and all family members for remembering me in prayers.

Special thanks to my Supervisor Dr. Omar Arif and Ex Supervisor and Mentor Dr. Shahzad Saleem who were always there to help, support and guide me during my thesis phase. I have learned a great deal from them. I feel so lucky while I count you people as my supervisors. Thank you

I would like to thank my committee members Dr. Sharifullah Khan; always big source of guidance and Dr. Hasan Tahir; the one who provided the exact directions to follow in different phases of thesis.

At the end I would like to give special thanks to my friends and fellows especially for those worth remembering moments we have enjoyed during our tenure in the campus.

**Farrukh Shabbir**

# **Dedication**

This dissertation is dedicated to

My beloved parents, brothers, sisters  
and friends.

## Abstract

In this modern era of rapidly increasing digitalization where most of the critical and important data resides on the storage of digital devices, out of which computers are the most commonly used devices on the planet. Computer users are higher in numbers among all digital devices and majority of them use Microsoft Windows being the user friendly Operating System (OS). Digital crimes are and will remain the major challenge associated with the latest developments in technology. Most of the threatening digital crimes belong to computer systems. Keeping in view the importance of computers in our lives and associated computer crimes, digital Investigations have become an important field and specially when Microsoft Windows; being most used OS is involved in the investigation. Windows OS registry is an important component which maintains record of almost all applications' activities and hence required to be digitally investigated. Windows Registry was introduced in Windows 3.1 and from there on registry is growing considerably in size with the evolution of Windows. The problem arises for digital investigators to find out the mouth-watering forensic artifacts from the provided huge volume of registry values. Finding such artifacts is a tedious task and takes a lot of time.

In order to solve huge registry puzzle, a Machine Learning (ML) based dynamic technique is introduced in this research which can automate extraction of relevant forensic artifacts from Windows Registry. Resulted technique will help in efficiently simplifying the Digital Investigations and makes Investigator's life simpler.

**Keywords:** *Digital Forensic Investigation, Windows Registry, Computer Forensics, Registry Forensics, Machine Learning (ML), Natural Language Processing (NLP), Windows Forensics*

# TABLE OF CONTENTS

## CHAPTER 1

<b>INTRODUCTION .....</b>	<b>1</b>
1.1 Background .....	1
1.2 Importance of Digital Evidence .....	1
1.3 Digital Devices and their Involvement in Digital Crimes .....	1
1.4 Most Widely Used Operating System: Windows .....	2
1.5 Windows Registry .....	3
1.5.1 Importance of Registry in Digital Forensics .....	4
1.6 Evolution of Windows Registry .....	4
1.7 Registry Analysis: A Challenge for Forensic Investigator .....	5
1.8 Problem Statement .....	5
1.9 Motivation .....	5
1.10 Solution Description .....	6
1.11 Thesis Flow .....	6

## CHAPTER 2

<b>WINDOWS REGISTRY ARCHITECTURE .....</b>	<b>7</b>
2.1 Introduction .....	7
2.2 Registry Hives .....	7
2.3 Registry Root Keys .....	8
2.4 Registry Hierarchical Structure .....	9
2.5 Registry Values .....	10
2.6 Registry Functional Overview .....	11

## CHAPTER 3

<b>LITERATURE REVIEW .....</b>	<b>12</b>
3.1 Introduction .....	12
3.2 Digital Forensic is a Separate Field .....	12
3.3 Digital Forensic as a Methodology .....	13
3.4 Windows Registry Forensics .....	13
3.5 Latest Studies in Registry Forensic .....	14
3.6 Machine Learning Techniques in Digital Forensics .....	17



3.7	Overview of Literature Review.....	20
-----	------------------------------------	----

## **CHAPTER 4**

### **MACHINE LEARNING: A HELPFUL RESOURCE .....21**

4.1	Introduction .....	21
4.2	Machine Learning: Use Cases .....	22
4.2.1	Video Surveillance .....	22
4.2.2	Cyber Security (Captchas).....	22
4.3	Importance of ML in Digital Forensics .....	23
4.4	Importance of ML in this Research .....	24
4.3.1	Word Embedding.....	24
4.3.2	Word2Vec .....	24

## **CHAPTER 5**

### **A MACHINE LEARNING BASED METHODOLOGY .....26**

5.1	Introduction .....	26
5.2	Proposed Methodology .....	26
5.2.1	Acquisition of Registry Hives .....	28
5.2.2	Parsing Registry Hives .....	28
5.2.3	Data Pre-Processing.....	29
5.2.4	Training ML Algorithm .....	29
5.2.5	Relations based Keyword Suggestions .....	30
5.2.6	Filtration .....	30

## **CHAPTER 6**

### **RESULTS.....33**

6.1	Introduction .....	33
6.2	Experiments & Evaluation .....	33
6.3	Results .....	37
6.2.1	Situation 1: List all Available Applications from Registry .....	37
6.2.2	Situation 2: Finding all Microsoft Word Documents from Registry .....	38
6.2.3	Situation 3: Discover Relations on the basis of Real Name of User .....	39

## **CHAPTER 7**

<b>CONCLUSION, LIMITATIONS AND FUTURE WORK.....</b>	<b>41</b>
7.1 Conclusion.....	41
7.2 Limitations .....	41
7.3 Future Work.....	41
 <b>REFERENCES .....</b>	 <b>43</b>

## LIST OF FIGURES

FIGURE 1. MOST VULNERABLE DEVICES [3].....	2
FIGURE 2. MARKET SHARE OF OPERATING SYSTEMS.....	3
FIGURE 3. MARKET SHARE OF OS VERSIONS.....	3
FIGURE 4. REGEDIT.....	8
FIGURE 5. STRUCTURE OF WINDOWS REGISTRY.....	9
FIGURE 6. VIRTUAL MACHINE FORENSIC ANALYSIS METHODOLOGY.....	17
FIGURE 7. ACCURACY OF ML ALGORITHMS.....	18
FIGURE 8. DESIGN MODEL FOR APPLICATION CLASSIFICATION.....	19
FIGURE 9. DECISION TREE LEARNING.....	21
FIGURE 10. ML BASED SURVEILLANCE CAMERA.....	22
FIGURE 11. CAPTCHA USING ML.....	23
FIGURE 12. CBOW WORD2VEC.....	25
FIGURE 13. SKIP GRAM WORD2VEC.....	25
FIGURE 14. METHODOLOGY WORK FLOW.....	27
FIGURE 15. ACQUISITION PHASE.....	28
FIGURE 16. PARSING PHASE.....	28
FIGURE 17. PRE-PROCESSING PHASE.....	29
FIGURE 18. ML TRAINING PHASE.....	29
FIGURE 19. KEYWORD SUGGESTIONS PHASE.....	30
FIGURE 20. FILTRATION PHASE.....	31
FIGURE 21. FUNCTIONALITY DIAGRAM - ALL PHASES COMBINED.....	32
FIGURE 22. RELATION BETWEEN RECALL AND PRECISION.....	37
FIGURE 23. CODE EXECUTION.....	38
FIGURE 24. LIST OF AVAILABLE APPLICATIONS IN REGISTRY.....	38
FIGURE 25. FINDING ALL MICROSOFT WORD DOCUMENTS FROM REGISTRY.....	38
FIGURE 26. EXPORTED RESULTS OF MICROSOFT WORD DOCUMENTS.....	39
FIGURE 27. REAL NAME BASED QUERYING.....	40
FIGURE 28. REAL NAME BASED QUERY RESULTS.....	40

## LIST OF TABLES

TABLE 1.	LONGITUDINAL UNDERSTANDING OF WINDOWS REGISTRY .....	4
TABLE 2.	REGISTRY KEYS AND VALUES INCREASING WITH EVOLUTION [10].....	5
TABLE 3.	REGISTRY HIVES FILES .....	7
TABLE 4.	REGISTRY ROOT KEYS.....	9
TABLE 5.	TYPES OF REGISTRY VALUES .....	10
TABLE 6.	DIGITAL EVIDENCE COLLECTION METHODOLOGY .....	13
TABLE 7.	FORENSIC ARTIFACTS COMPARISON: WINDOWS 8.1 (LEFT) AND ADDITIONAL IN WINDOWS 10 (RIGHT).....	14
TABLE 8.	COMPARISON OF TOOLS ACQUISITION.....	14
TABLE 9.	COMPARISON IN CASE OF USB DEVICES.....	15
TABLE 10.	COMPARISON IN CASE OF MOBILE DEVICES .....	16
TABLE 11.	DISTRIBUTION OF INSTANCES IN TRAINING DATASET .....	18
TABLE 12.	EVALUATION RESULTS .....	34

## INTRODUCTION

### 1.1 Background

Digitalization is growing rapidly in the world and is a well-known practice which is adopted by most of the people and organizations. Development in digitalization resulted in invention of numerous digital tools and devices. Digitalization has become a requirement in our daily schedules and most of the individuals are seriously attached with digital devices to fulfill such requirement. Keeping in view the ever growing usage of digital devices, digital criminals are becoming popular as well due to their nature of finding and exploiting vulnerabilities to perform digital crimes. Activities of digital criminals led to the existence of digital investigations to cope up with digital crime cases. Digital crimes have grown exponentially in the recent past and most of them are very severe i.e. ATM Jackpotting is a technique in which ATM machines are triggered to spit cash out of the ATM machines. Digital investigation is a way [1] which can be adopted in the cases where digital crimes are involved and as a result evidences are collected which can be useful in the court of law.

### 1.2 Importance of Digital Evidence

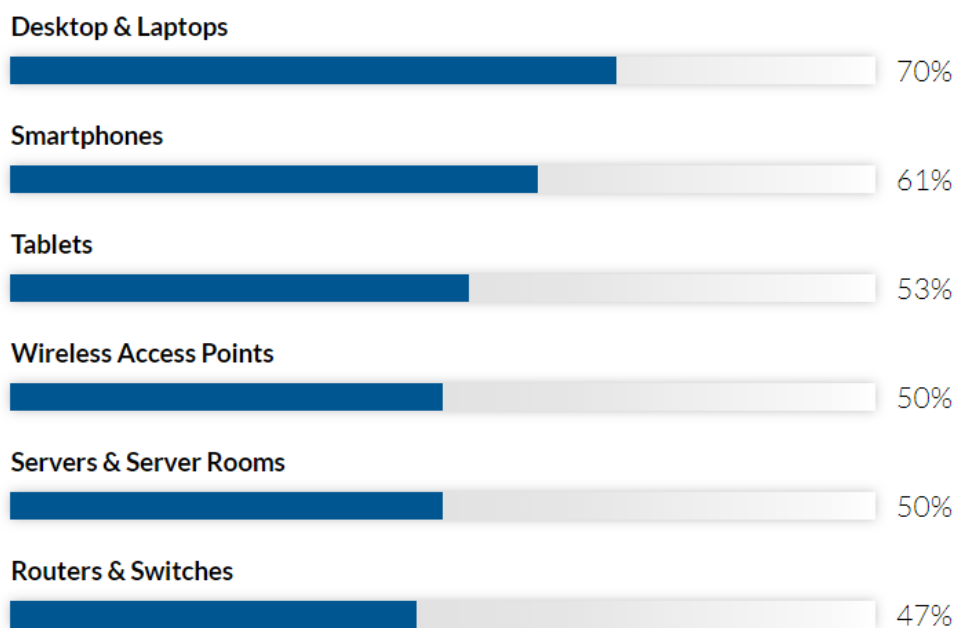
Digital evidence is as important as an eye witness in the proceedings of court. The way an eye witness can be bribed or killed, a digital evidence can also be forged or even destroyed. In fact, digital evidence can be tempered with more ease due to its fragile [2] characteristics. Hence, digital evidence require more care as compared to physical evidence and a mechanism should be adopted to avoid its tempering. Moreover, chain of custody and its associated requirements are to be handled with care.

### 1.3 Digital Devices and their Involvement in Digital Crimes

Digital devices are prone to attacks which cause occurrence of digital crime. Almost every digital device is vulnerable and can be used to execute a criminal activity

which leads to the birth of digital evidence. Figure 1 illustrates the percentages of most vulnerable digital devices to digital crimes, which depicts that computers are the most vulnerable devices and are targeted by most of the cyber criminals. As a result computers are needed to be analyzed to harvest valuable information about the criminal activities performed either directly on them or by using them as a proxy or pass-through.

**Most Vulnerable Devices to Cybercrimes**  
IT pros consider the following as having the highest level of risk to hacks, breaches, and other cyber threats:



Source: AT&T 2018 Cybersecurity Report

Designed by  FinancesOnline

FIGURE 1. MOST VULNERABLE DEVICES [3]

#### 1.4 Most Widely Used Operating System: Windows

Microsoft windows is the most widely used Operating System (OS) due to its popularity and user friendliness. Among all available operating systems Microsoft Windows has maximum chunk of market share [4] as 80% plus of the computer are Windows based as per the stats in Figure 2.

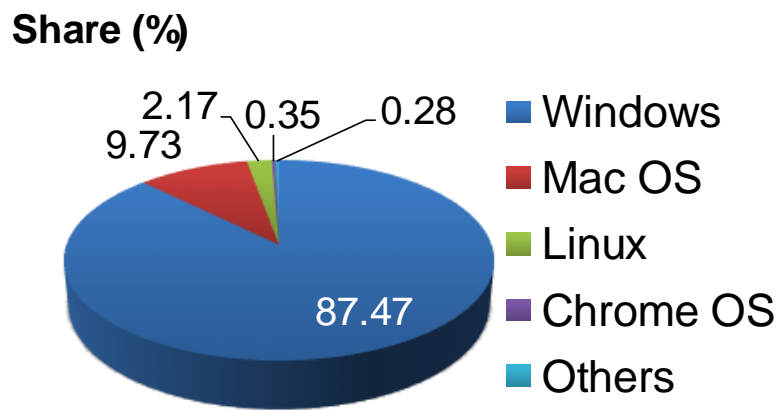


FIGURE 2. MARKET SHARE OF OPERATING SYSTEMS

Between versions of operating system Microsoft Windows 10 and Windows 7 are holding the most of the market share with 40% and 38% as shown in Figure 3. Due to its popularity and being widely used operating system, Windows is targeted in this study for introducing a new level of research by involving Machine Learning techniques in the field of Windows Registry Forensics.

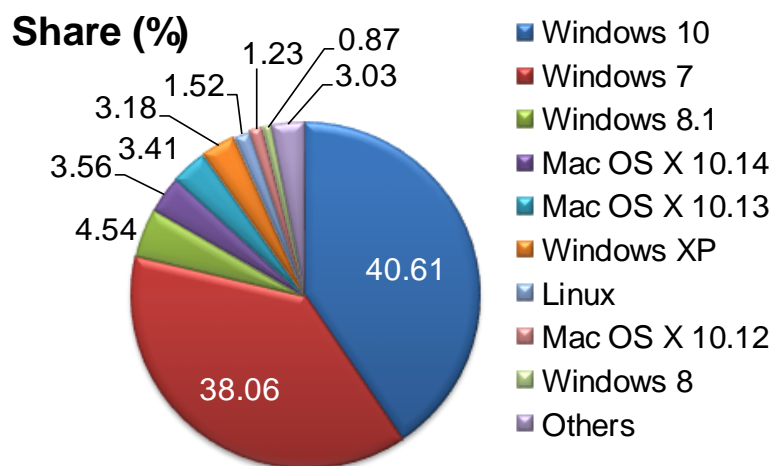


FIGURE 3. MARKET SHARE OF OS VERSIONS

### 1.5 Windows Registry

Linux keeps configuration data related to applications in files, similarly, Windows maintains registry for the same purpose. Windows registry is a database [5] which keeps system configurations, applications' configurations and users'

configurations data which provides a great deal of information to a forensic investigator. More on Windows registry is explained in the next chapter.

### 1.5.1 Importance of Registry in Digital Forensics

As mentioned above, registry stores a lot of valuable information about users and applications activities which can be very helpful in registry forensics [6][7], it also contains attributable artifacts, sometimes even if a program is removed from the windows. A user can install anti-forensic tool i.e. data shredders, and then uninstall it. But the remnants of used anti-forensic tools will remain in the registry and will be helpful in finding out the occurrence of such activity [8]. Most of the computer crimes are executed on or using Windows OS being the friendliest operating system. Thus, windows registry is an important area for collecting forensically sound artifacts during digital investigation procedures.

### 1.6 Evolution of Windows Registry

A longitudinal understanding of registry features evolving throughout different versions of Microsoft Windows are presented by Avinash Singh et al. [9] and are depicted in Table 1.

TABLE 1. LONGITUDINAL UNDERSTANDING OF WINDOWS REGISTRY

Hive Files	Windows 95	Windows 98	Windows 2000	Windows XP	Windows VISTA	Windows 7	Windows 8	Windows 10
BCD	-	-	-	-	✓	✓	✓	✓
DRIVERS	-	-	-	-	-	-	-	✓
SAM	-	-	✓	✓	✓	✓	✓	✓
SECURITY	-	-	✓	✓	✓	✓	✓	✓
SOFTWARE	-	-	✓	✓	✓	✓	✓	✓
SYSTEM	-	-	✓	✓	✓	✓	✓	✓
DEFAULT	-	-	✓	✓	✓	✓	✓	✓
COMPONENTS	-	-	-	-	✓	✓	✓	✓
NTUSER.DAT	-	-	✓	✓*	✓*	✓*	✓*	✓*
USRCLASS.DAT	-	-	-	✓*	✓*	✓*	✓*	✓*
SYSTEM.DAT	✓	✓	-	-	-	-	-	-
USER.DAT	✓	✓	-	-	-	-	-	-
Policy.pol	-	✓	-	-	-	-	-	-

Items marked with a (\*) may have more than one location for the file



## 1.7 Registry Analysis: A Challenge for Forensic Investigator

Registry data is ever growing with the evolution of Windows operating system versions and hence registry Keys and Values by default are increasing significantly [10] as depicted in Table 2.

TABLE 2. REGISTRY KEYS AND VALUES INCREASING WITH EVOLUTION [10]

Registry Hive	Windows 10		Windows 7		% Difference	
	Keys	Values	Keys	Values	Keys	Values
<b>HKLM</b>	568,162	343,200	354,553	217,193	+160%	+158%
<b>HKCR</b>	187,458	161,053	113,642	94,597	+165%	+170%
<b>HKU</b>	29,505	13,806	7,182	2,554	+411%	+540%
<b>HKCU</b>	10,563	5,237	4,486	1,906	+235%	+275%

## 1.8 Problem Statement

“Digital Forensic case resolution requires forensically sound artifacts. Conventionally such artifacts are collected manually which is a tedious and time consuming task and produces unwanted delay in digital investigations because of the enormous amount of registry keys and values, which are continuously growing with the development of new Windows versions.”

## 1.9 Motivation

Currently, registry forensics can be performed with the help of multiple tools and approaches but most of them requires manual interaction with a huge amount of registry data most of which is not useful in some cases. Thus, it is annoying for a forensic investigator to use these procedures based on manual tracking of valuable registry artifacts. Aim is to help the digital investigations in a dynamic way by introducing a technique which will provide appropriate and most relevant registry results. It will decrease the extra time and effort of digital forensic investigator and will improve the digital investigation process.

### 1.10 Solution Description

“To replace tedious and time consuming manual digital investigation procedures, an efficient heuristic tool is required to automatically collect forensically sound registry artifacts even across any version of windows operating system.”

This research is carried out by keeping in mind the huge amount of registry data which is increasing exponentially with the evolution of Windows. A technique is proposed to manage the burden of enormous registry keys and values to help Windows registry forensic analysts. Windows registry is a sea carrying key evidence which can resolve the mystery of digital investigations. Results of Machine Learning based technique are compiled to evaluate the usage of research.

### 1.11 Thesis Flow

There are seven chapters in this thesis. Distribution of chapters is given below:-

**Chapter 1:** Provides the Introduction of the research

**Chapter 2:** Explains Microsoft Windows Registry Architecture

**Chapter 3:** Discusses the related work in the same field

**Chapter 4:** Provides an overview about importance of Machine Learning

**Chapter 5:** Proposes a Machine Learning based technique for automatic collection of forensically sound artifacts

**Chapter 6:** Results after applying the ML based technique

**Chapter 7:** Conclusion, Limitations and Future work.

WINDOWS REGISTRY ARCHITECTURE

2.1 Introduction

Microsoft defined Windows Registry in the fifth edition of computer dictionary as [11] a hierarchical central database inside Windows operating system which keeps necessary configuration data of software, hardware and users. The detailed Windows Registry architecture is well-defined by Microsoft in [12].

2.2 Registry Hives

There are a total of six Registry hive files along with supporting files against each as shown in Table 3. Hive file contains *keys*, *subkeys* and *values*. Hive files and supporting files are kept at *Windows/System32/Config* folder apart from *NTUser.dat*. *NTUser.dat* contains very valuable configuration settings related to a particular user. Each user has its own *NTUser.dat* file which is stored in user’s profile. Data stored in *NTUser.dat* can be directly attributed to the activities performed by a specific user.

TABLE 3. REGISTRY HIVES FILES

Registry Hive	Supporting Files
HKEY_LOCAL_MACHINE\SAM	Sam, Sam.log, Sam.sav
HKEY_LOCAL_MACHINE\Security	Security, Security.log, Security.sav
HKEY_LOCAL_MACHINE\Software	Software, Software.log, Software.sav
HKEY_LOCAL_MACHINE\System	System, System.alt, System.log, System.sav
HKEY_CURRENT_CONFIG	System, System.alt, System.log, System.sav, Ntuser.dat, Ntuser.dat.log
HKEY_USERS\DEFAULT	Default, Default.log, Default.sav

### 2.3 Registry Root Keys

Windows includes a built-in editor to access registry by using command *regedit* at command prompt. Current Registry data including keys and values can be viewed in registry editor. Registry root keys can be visible in the registry editor as shown in Figure 4 and each root key is explained in details in Table 4 below [13].

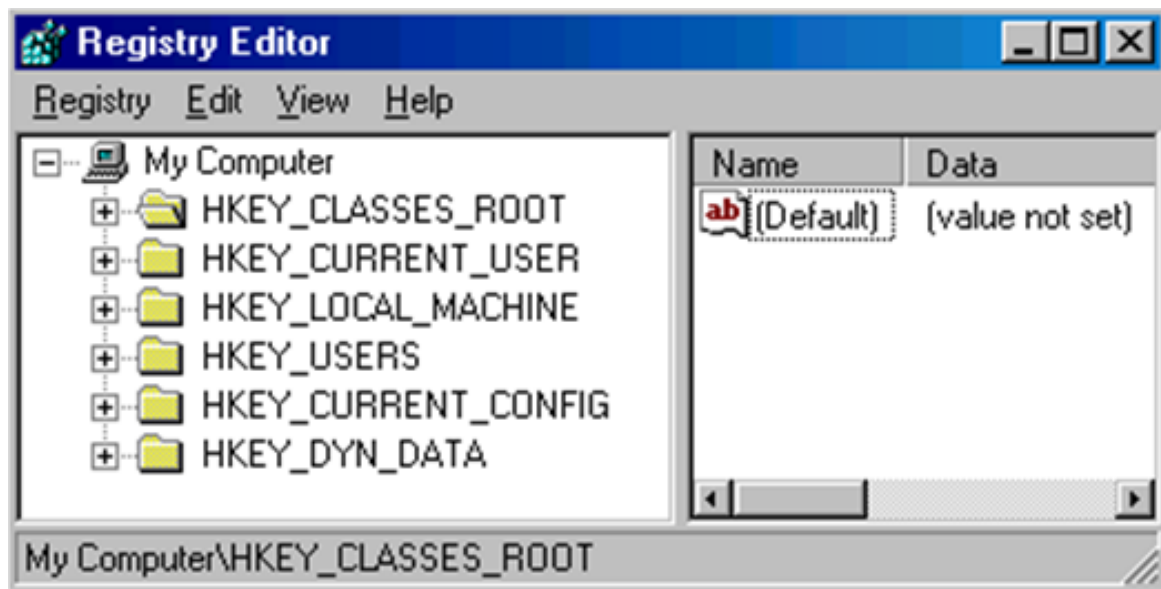


FIGURE 4. REGEDIT

TABLE 4. REGISTRY ROOT KEYS

Root Key	Description
<b>HKCR</b> (HKEY_CLASSES_ROOT)	Describes file type, file extension, and OLE information.
<b>HKCU</b> (HKEY_CURRENT_USER)	Contains user who is currently logged into Windows and their settings.
<b>HKLM</b> (HKEY_LOCAL_MACHINE)	Contains computer-specific information about the hardware installed, software settings, and other information. The information is used for all users who log on to that computer. This key, and its subkeys, is one of the most frequently areas of the registry viewed and edited by users.
<b>HKU (HKEY_USERS)</b>	Contains information about all the users who log on to the computer, including both generic and user-specific information.
<b>HKEY_CURRENT_CONFIG (HKCC)</b>	The details about the current configuration of hardware attached to the computer.
<b>HKDD (HKEY_DYN_DATA)</b>	Only used in Windows 95, 98, and NT, the key contained the dynamic status information and plug and play information. The information may change as devices are added to or removed from the computer. The information for each device includes the related hardware key and the device's current status, including problems.

2.4 Registry Hierarchical Structure

Windows registry keeps information about configurations related to users, applications and devices in a hierarchical format (a tree-based structure) [5] as shown in Figure 5.

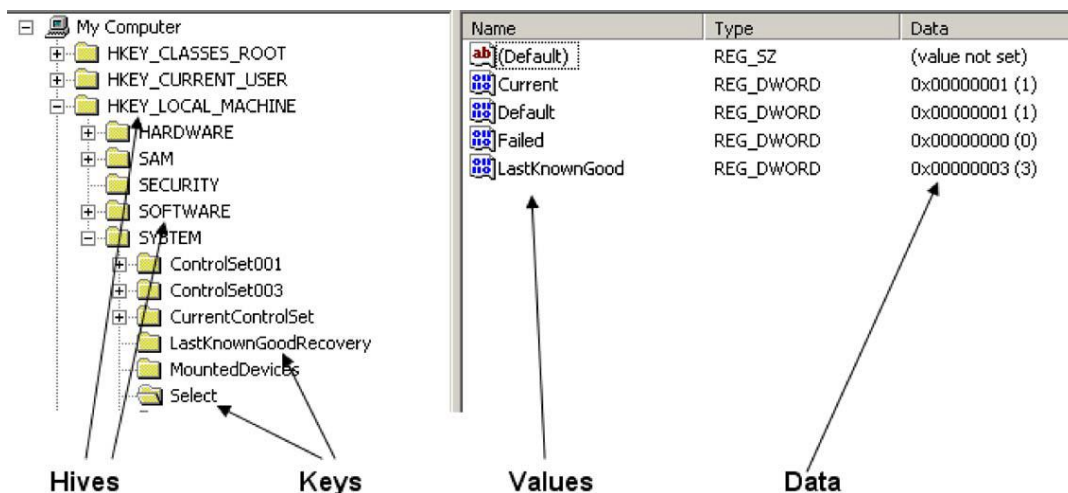










FIGURE 5. STRUCTURE OF WINDOWS REGISTRY

## 2.5 Registry Values

Registry value holds the most important data as we traverse through the hierarchical keys to see the Registry value stored at the last node in registry editor. On finding a value, it is important to understand the format which describes the way a value is keeping data in it. Description of different type of registry value formats are mentioned in Table 5 [13].

TABLE 5. TYPES OF REGISTRY VALUES

Icon	Type	Name	Description
		<b>Closed key</b>	Like the folders seen in Windows Explorer. These keys are what contain the registry subkeys mentioned below.
		<b>Open key</b>	When a key is opened, the icon changes to an expanded or open folder and displays all its contents and any additional subkeys.
	REG_SZ	<b>String value</b>	Allows for any <a href="#">string</a> value to be defined on a single line, such as a file path, and is the most commonly found subkey in the registry.
	REG_MULTI_SZ	<b>String array value</b>	Any multi-line string value.
	REG_EXPAND_SZ	<b>Expanded string value</b>	Contains a string with <a href="#">environmental</a> or <a href="#">system variables</a> that need to be expanded. For example, c:\%windir%\example.exe could be the same as C:\windows\example.exe.
	REG_BINARY	<b>Binary value</b>	Allows for attributes to be defined in <a href="#">binary</a> as either on or off (0 or 1).
	REG_DWORD	<b>DWORD value</b>	Similar to the binary value, but capable of values being defined in either 32-bit <a href="#">decimal</a> or <a href="#">hex</a> .
	REG_QWORD	<b>QWORD value</b>	Like the DWORD, but stored as a 64-bit value.

## 2.6 Registry Functional Overview

A default form of registry takes birth whenever a Windows is installed. Each version of Windows has its own default registry configurations but most of the keys and values are same except few additional one which are keep on increasing with every latest Windows operating system version. These default registry information keeps the basic data required for the running of Windows. After successful installation, a user started using the windows and performs multiple activities i.e. accessing file explorer, installation of different applications, using internet, inserting USBs etc. Each of these activities have their relevant changes occurrences inside registry and most of them belongs to *NTUser.dat* because this hive file is related to user's activities. Those changes or usages of a particular user may be kept in the registry even for a long duration. It also includes information about both installed and even uninstalled applications. Thus, providing a great deal in digital investigations involving Windows operating systems. But tracking the forensically sound registry entries or changes is still the major challenge for a forensic investigator.

### LITERATURE REVIEW

#### 3.1 Introduction

In this modern age, most of the physical data is replaced with the digital data, physical documents are decreased and digital documents are increased in numbers. Digital data is very frequently travelling and most of it is residing on digital devices. Major portion of such digital data is residing on computers including Desktops and Laptops. But on the other hand, criminal minds are part of each type of community which includes criminals from digital world as well and are called cyber criminals. Cyber criminals are of different type but their main targets are the digital devices including computers. Purpose of cyber criminals is to get access of confidential data residing on computers and then they can steal, destroy or modify the data. Most of the people in digital world are innocent and do not know much about activities of cyber criminals. But once a crime has occurred, digital investigation is essential to solve such criminal cases. A lot of work is required for increasing the efficiency of digital investigations and ultimately solving the criminal cases with ease. This chapter covers the previously performed studies in the field of registry forensics as well as in machine learning based digital forensics as discussed in the following paragraphs.

#### 3.2 Digital Forensic is a Separate Field

It is important to understand that Digital Forensic is itself a separate field in science which has its own characteristics, same is highlighted by Darek Bem et al. [14]. There are always some vulnerabilities which are not addressed and lead to the occurrence of digital crimes. There should be well documented policies and procedures for digital investigations covering all the gaps to avoid or address occurrences of digital crimes. Recent and upcoming requirements of digital forensics are also mentioned in the study which includes the chain of custody issues as well.



### 3.3 Digital Forensic as a Methodology

A methodology is defined for each of the digital forensic processes i.e. Extraction, Identification and Analysis by Peter Cisar et al. [15] to sort out the issue of multiple digital forensic methodologies in practice at that time. Sabah Al-Fedaghi et al. [16] suggested an intellectual model for digital forensic investigations including operations i.e. arrive, accept, process, release, create and transfer. An autopsy tool for maintaining chain of custody to keep the evidence faithful is introduced to help digital forensic process. The tool is applicable for Kali Linux platform and uses Message Digest 5 (MD5) hashes to achieve the required authenticity of evidence. Table 6 shows the different phases of the study.

TABLE 6. DIGITAL EVIDENCE COLLECTION METHODOLOGY

No. of Phase	Digital Evidence Collection Methodology
Phase 1	Confirmation
Phase 2	System Explanation
Phase 3	Proof Acquisition
Phase 4	Timeline Evaluation
Phase 5	Mass Media Artifact Evaluation
Phase 6	Sting Byte Search
Phase 7	Data Collection
Phase 8	Reporting Result

### 3.4 Windows Registry Forensics

Diana Hinteá et al. [17] discussed the changes regarding registry data after comparing Windows 10 registry with Windows 8.1 registry. They listed the additional registry artifacts as shown in Table 7 which are introduced in Windows 10 and were not in Windows 8.1.

TABLE 7. FORENSIC ARTIFACTS COMPARISON: WINDOWS 8.1 (LEFT) AND ADDITIONAL IN WINDOWS 10 (RIGHT)

<i>Artefacts</i>	<i>Artefacts</i>
Prefetch files	Notification centre
Event logs	New start menu
USB activity	Cortana
Recycle bin	Windows 10 applications
OneDrive	OneDrive data
Internet Explorer	Frequent folders
LNK files	
Thumbnails	

Windows registry forensics are performed by many researchers but most of the research was targeted on the basis of some application or software. Similarly Raihana Md Saidi et al. [18] performed registry forensics to find out the forensic artifacts related to Virtual Network Computing (VNC) Software and Keylogger softwares. Attackers commonly use such applications to compromise a computer and get valuable information about the activities performed on the computer. Remnants of such application are gathered in this study and presented as results. Muhammad Nur Faiz et al. [19] worked in the area of Live Forensics and compared multiple memory acquisition tools as listed in Table 8. Forensics performed on volatile RAM data which is disappeared whenever the computer is shutdown is called Live Forensics.

TABLE 8. COMPARISON OF TOOLS ACQUISITION

Tools	Memory Usage (Mb)	Processing Time (second)	Registry Key	DLL
FTK Imager	117	198.65	59	270
Belka RAM Capturer	18	186.22	9	56
Magnet RAM Capture	76	220.24	98	285
Dumplt	10	185.6	4	44
Memoryze	13	184.54	7	71

### 3.5 Latest Studies in Registry Forensic

Keeping in view the area of windows registry forensics, most recently USB and Mobile devices activities were monitored on different versions of Windows OS by

Ayesha Arshad et al. [20]. The research is based on collecting the event logs and registry data on the activities of insertion and removal of USB and smart phones. Artifacts relevant to insertion of USB device is presented in Table 9 and artifacts related to insertion of smart phones are shown in Table 10. But registry is only monitored for external devices not overall.

TABLE 9. COMPARISON IN CASE OF USB DEVICES

	Key/Subkey	Win 7	Win 8	Win 10
First insertion time from DeviceClasses key in System Hive	10497b1b-ba51-44e5-8318-a65c837b6661	✓	✓	✓
	53f56307-b6bf-11d0-94f2-00a0c91efb8b	✓	✓	✓
	53f5630d-b6bf-11d0-94f2-00a0c91efb8b	✓	✓	✓
	65a9a6cf-64 cd-480b-843e-32c86e1ba19f	✓		
	6ac27878-a6fa-4155-ba55-f95f491d4f33	✓	✓	✓
	7f108a28-9833-4b3b-b780-2c6b5fa5c062		✓	✓
	7fcc86c-228a-40ad-8a58-f590af7bfdce		✓	✓
	a5dcbf10-6530-11d2-901f-00c04fb951ed	✓	✓	✓
	EEC5AD98-8080-425f-922A-DABF3DE3F69A	✓	✓	✓
	f33fdc04-d1ac-4e8e-9a30-19bbd4b108ae	✓	✓	✓
First insertion time from System Hive Under USBSTOR Property Key	0003	✓	✓	✓
	000A		✓	✓
	0064	✓	✓	✓
	0065	✓	✓	✓
Last insertion time from System Hive Under USBSTOR Property Key	0066		✓	✓
	0067		✓	✓
First insertion time from System Hive under USB Key	VID_{VendorID}&PID_{ProductID}\{SerialNo}\DeviceParameters e5b3b5ac-9725-4f78-963f-03dfb1d828c7			✓
Last insertion time from System Hive under USB Key	VID_{VendorID}&PID_{ProductID}\{SerialNo}\	✓	✓	✓
First insertion time from Software Hive	Microsoft\Windows Portable Devices\Devices \SWD#WPDBUSENUM#_??_ USBSTOR#DISK&VEN_{VendorName} &PROD_{ProductName}&REV_PMAP#\{SerialNo#\{53F56307-B6BF-11D0-94F2-00A0C91EFB8B}	✓	✓	✓

TABLE 10. COMPARISON IN CASE OF MOBILE DEVICES

	Key/Subkey	Win 7	Win 8	Win 10
First insertion Time from DeviceClasses key in System Hive	10497b1b-ba51-44e5-8318-a65c837b6661	✓	✓	✓
	6ac27878-a6fa-4155-ba55-f95f491d4f33	✓	✓	✓
	6bdd1fc6-810f-11d0-bec7-08002be2092f	✓	✓	✓
	a5dcbf10-6530-11d2-901f-00c04fb951ed	✓	✓	✓
	EEC5AD98-8080-425f-922A-DABF3DE3F69A	✓		
	f33fdc04-d1ac-4e8e-9a30-19bbd4b108ae	✓	✓	✓
First insertion time from System Hive Under USB Property Key	0003	✓	✓	✓
	0007		✓	✓
	0008		✓	✓
	0009		✓	✓
	000A		✓	✓
	0064	✓	✓	✓
	0065	✓	✓	✓
Last insertion time from System Hive Under USB Property Key	0066		✓	✓
Last removal time from System Hive Under USB Property Key	0067		✓	✓
First insertion time from System Hive under USB Key	VID_[VendorID]&PID_[ProductID]\{SerialNo}\DeviceParameters \\e5b3b5ac-9725-4f78-963f-03dfb1d828c7			✓
Last insertion time from System Hive under USB Key	VID_[VendorID]&PID_[ProductID]\{SerialNo}\	✓	✓	✓

Forensic activities related to illegal download of copyright data using torrent clients and data stolen activities using USBs were highlighted by Hasan Binjuraid et al. [21]. Virtualization is very famous in recent years and virtual machines are most widely used instead of physical machines. Forensic artifacts collection of Virtual Machine (VM) is carried out by Erfan Wahyudi et al. [22]. The research is performed on VirtualBox, a tool used for virtualization and Regshot tool is used to get the registry changes before creation of VM and after deletion of VM. The adopted methodology is shown in Figure 6.

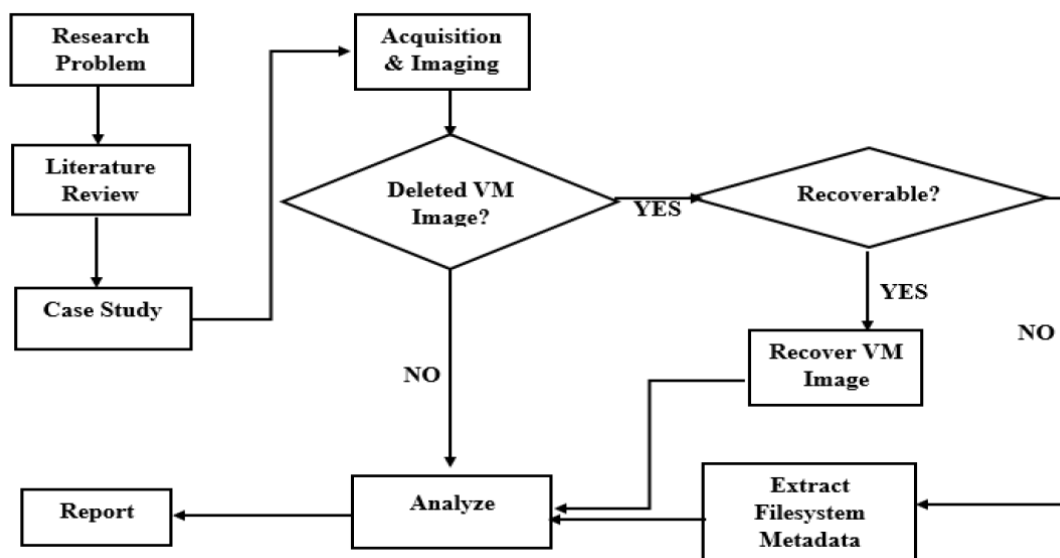


FIGURE 6. VIRTUAL MACHINE FORENSIC ANALYSIS METHODOLOGY

### 3.6 Machine Learning Techniques in Digital Forensics

To solve the challenge of Big Data in the field of forensics, machine learning techniques can be used as one of the solutions. Rami Mustafa A. Mohammad et al. [23] used historical file system data and converted it in the form of dataset to be used in training the multiple machine learning algorithms. Moreover, ML algorithms are then used to track down the evidence of different applications manipulating the files. Table 11 shows the training dataset distribution instances and Figure 7 is depicting the accuracy percentages of different ML algorithms used in the study.

TABLE 11. DISTRIBUTION OF INSTANCES IN TRAINING DATASET

No	Application	Number of instances
1	MS-Word	4983
2	MS-Excel	4665
3	MS-PowerPoint	3764
4	MS-Access	4210
5	NetBeans	3692
6	Adobe Acrobat Reader	3687
7	VLC media player	6013
8	WEKA	2955
9	MS-Paint	2058
10	Internet Explorer	6501
	<b>Total</b>	<b>42,528</b>

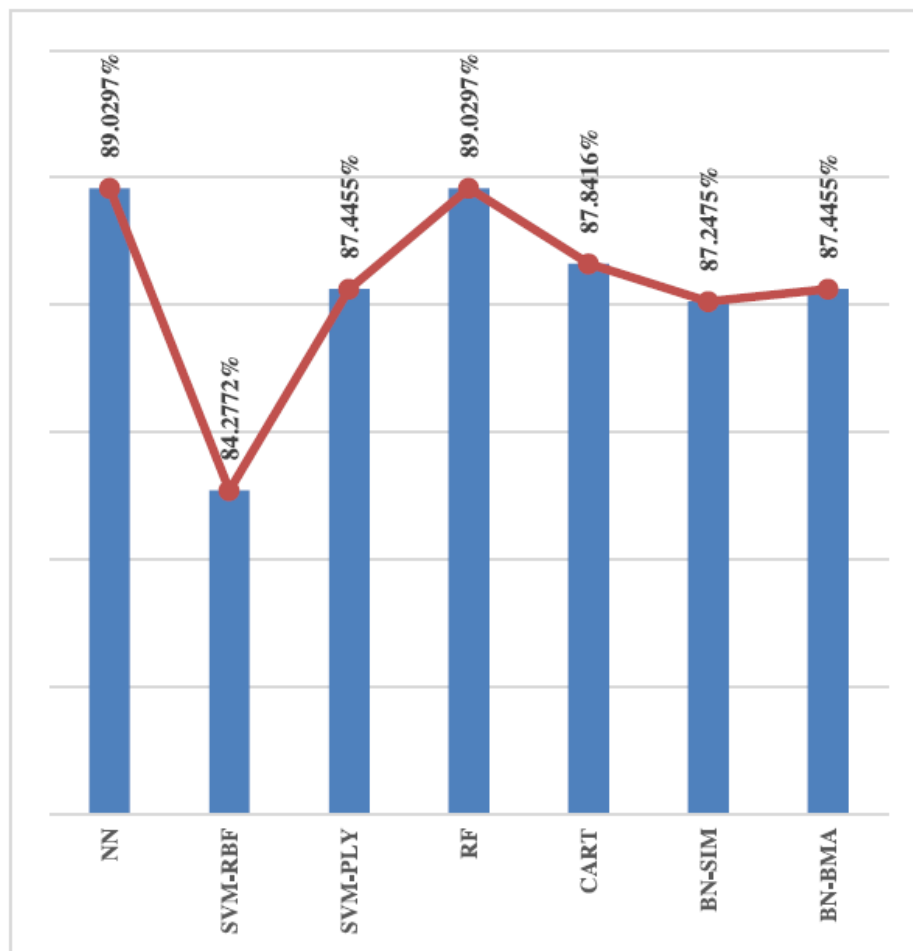


FIGURE 7. ACCURACY OF ML ALGORITHMS

Edem Inang Edem et al. [24] proposed malware analysis technique on the basis of machine learning algorithms to support digital forensics relevant to finding artifacts related to malware occurrences in a computer. The proposed methodology provides a dynamic solution which is helpful in digital investigations on malware analysis. A framework for constructing a post-event timeline with the help of neural networks is introduced by M. N. A. Khan et al. [25]. Neural networks are trained to classify the data of different applications present in the Disk Image taken for forensic investigations. Figure 8 shows the design model for classifying the applications data from the provided disk image.

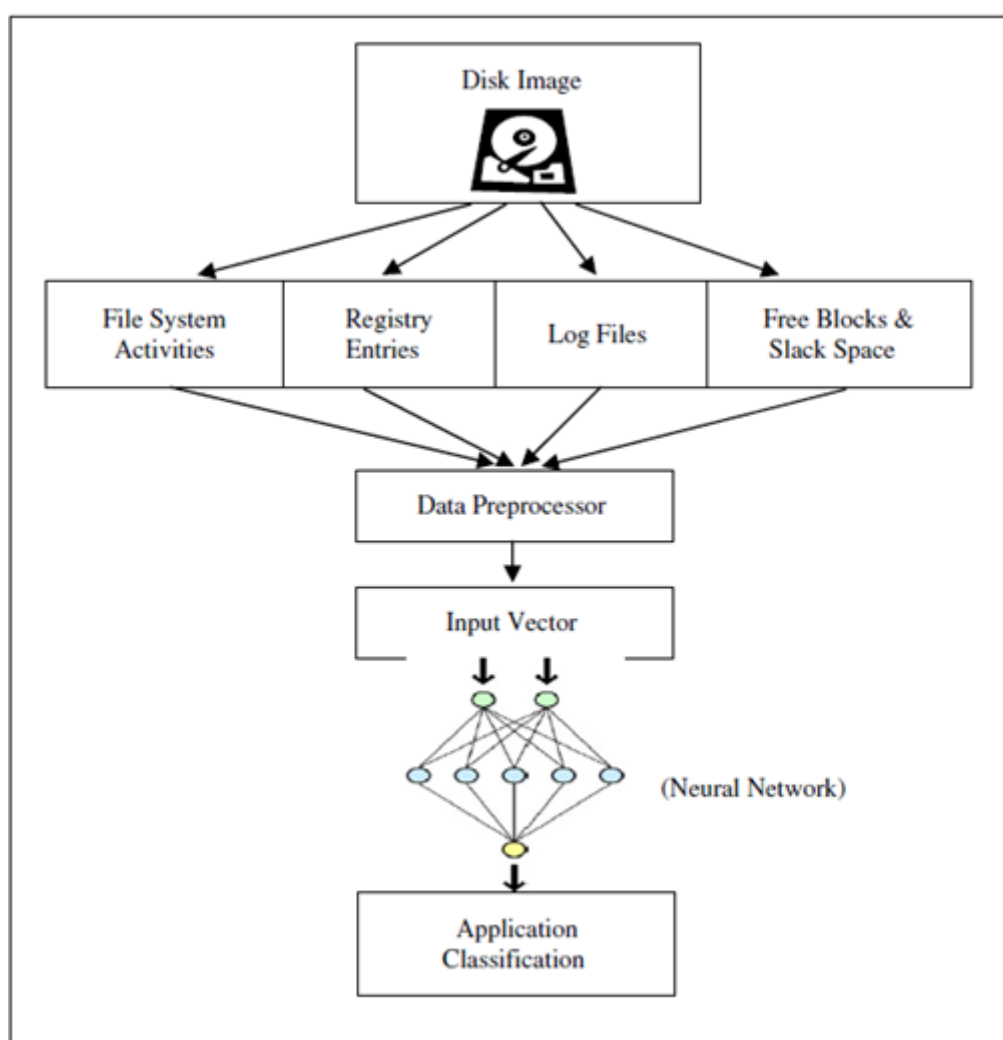


FIGURE 8. DESIGN MODEL FOR APPLICATION CLASSIFICATION

However, none of the machine learning based study has provided a solution to get important forensic artefacts after applying ML on Windows registry data.

### **3.7 Overview of Literature Review**

There is a handsome amount of work carried out in the field of Digital Forensics and some of which includes Windows registry forensics. Furthermore, there are few studies which machine leaning as a resource to provide solution for Big Data Forensics. However, all these studies are helpful in digital investigations but application of machine learning technique directly on Windows registry is still not addressed. In this research, a machine learning based technique is introduced to automatically analyze the huge amount of windows registry keys and values. With the help of such technique, digital investigator will be able to automatically achieve the required evidence helpful in solving a digital investigation case.



MACHINE LEARNING: A HELPFUL RESOURCE

4.1 Introduction

In this chapter a brief introduction of machine learning is provided. Machine Learning (ML) is a child field of Artificial Intelligence. The purpose of ML is to solve the problem of analyzing Big Data by understanding the format of data and convert it into some cluster form which can easily be understood by the users and hence utilized for further necessary actions. Nowadays, ML is an important area of computer sciences [26] in which ML algorithms are used to train the computers and then trained computers are used to perform different type of processes i.e. decision making on the basis of trained input data. Decision making results are in the form of a cluster of similar objects which are somehow related to each other and these results are very important for the users to get the required outputs. For example decision tree shown in Figure 9 classifies the data for days weather conditions if it is raining or not.

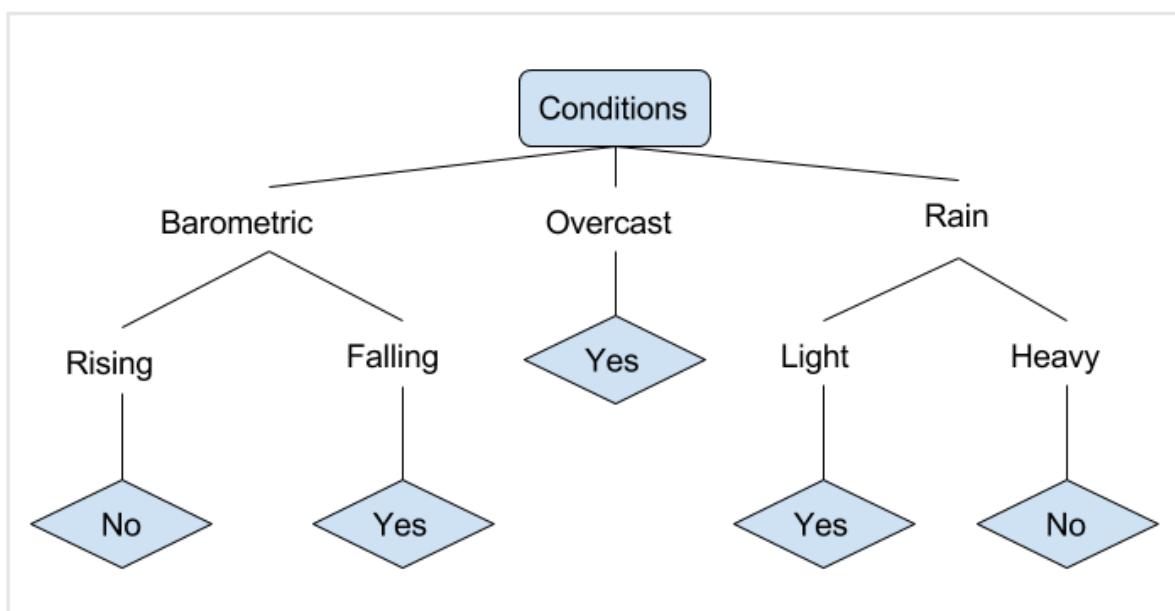


FIGURE 9. DECISION TREE LEARNING

## 4.2 Machine Learning: Use Cases

Machine Learning has plenty of use cases but keeping in view the aspect of this research only security related use cases are discussed. Recently, security industry is also inspired by the benefits of Machine Learning and is applied in many areas of security i.e. in traditional security, security guards were there to note down the vehicle numbers but now ML can be used to record the same automatically. Few use cases of ML in security are [27]:

### 4.2.1 Video Surveillance

Manually monitoring the video surveillance cameras is time consuming and is a tiring task. A solution is available in the world which performs security relevant tasks i.e. detecting intruders by using the surveillance cameras (shown in Figure 10) with built in machine learning features.



FIGURE 10. ML BASED SURVEILLANCE CAMERA

### 4.2.2 Cyber Security (Captchas)

Most of us are familiar with Captcha tool as shown in Figure 11. Captcha is also using machine learning algorithms to secure the websites from any illegal use by applying security based on recognizing a human being by asking

for finding the pictures related to traffic signals, cars etc. The purpose behind captcha is to mitigate Denial of Service (DoS) attacks.

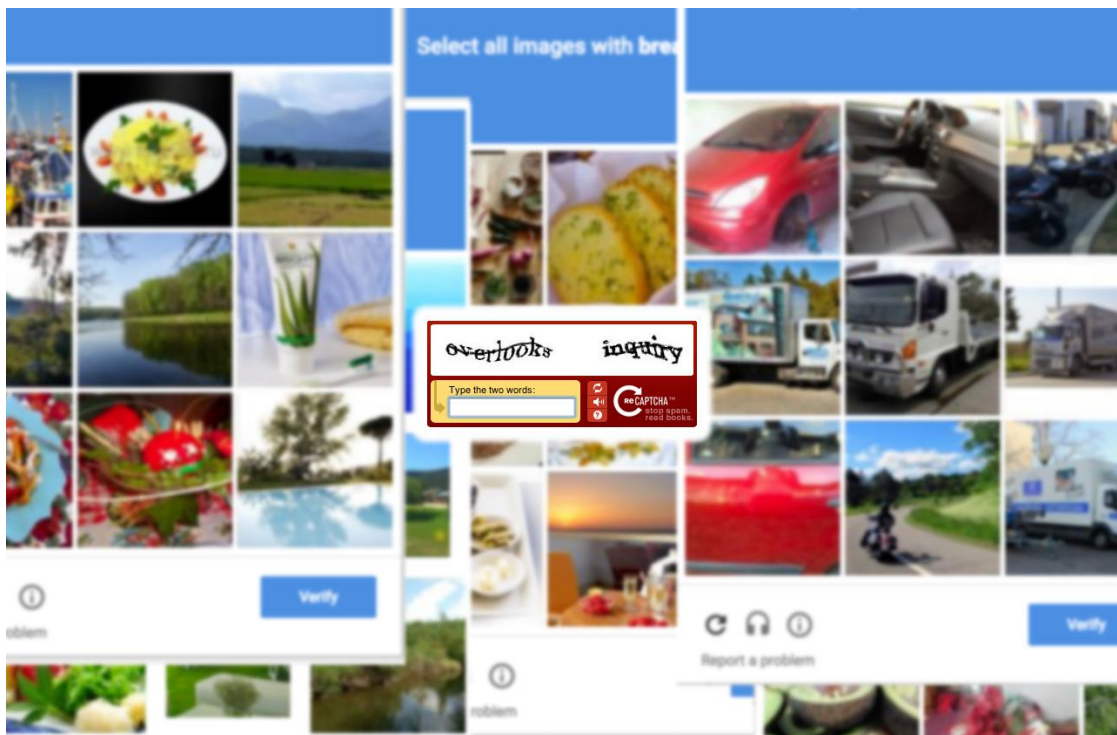


FIGURE 11. CAPTCHA USING ML

### 4.3 Importance of ML in Digital Forensics

Digital forensic data is growing day by day in size which makes it very difficult for forensic investigators to analyse the huge amount of data and solve the digital crime case. This increase in data is also causing delays in the digital forensic investigations. Reza Montasari et al. [28], introduced the latest challenges faced during the digital investigations, one of which is the Big Forensic Data (BFD). BFD is the huge amount of forensic data which needs to be analysed and relevant artifacts are required to be extracted. The study proposes the use of machine learning based techniques for solving the hurdle of BFD. They also defined Digital Forensic as a Service (DFaaS) which needs to be provided in order to solve the forensic cases more efficiently and quickly.

## 4.4 Importance of ML in this Research

Keeping in view the huge volume of registry data which is keep on increasing, now is the time to apply Machine Learning based technique to cope up with the growth of available Big Forensic Data (BFD). Same is applied in this research with the help of word embedding based ML algorithm.

### 4.3.1 Word Embedding

Word embedding is used in Natural Language Processing. It is a technique used to learn the features from given text and on the basis of those features provides opinions. It is the most famous illustration for finding document vocabulary. Relations of a word with other words within document or a given data a found and shown as results. It helps us in finding the relations of a particular word i.e. milk has relations with butter, yogurt and curd.

### 4.3.2 Word2Vec

It is a technique used to map the words to numerical vectors. It is used for language modeling which generates word embedding. Word2Vec consist of input, hidden and output layers. There are two ways of utilizing Word2Vec:

- a) **Continuous Bag of Words (CBOW).** On the basis of multiple context words CBOW Word2Vec predicts a single word as output. Hidden layer is used to present the number of dimensions for representation of output word. Figure 12 shows the workflow of CBOW Word2Vec.

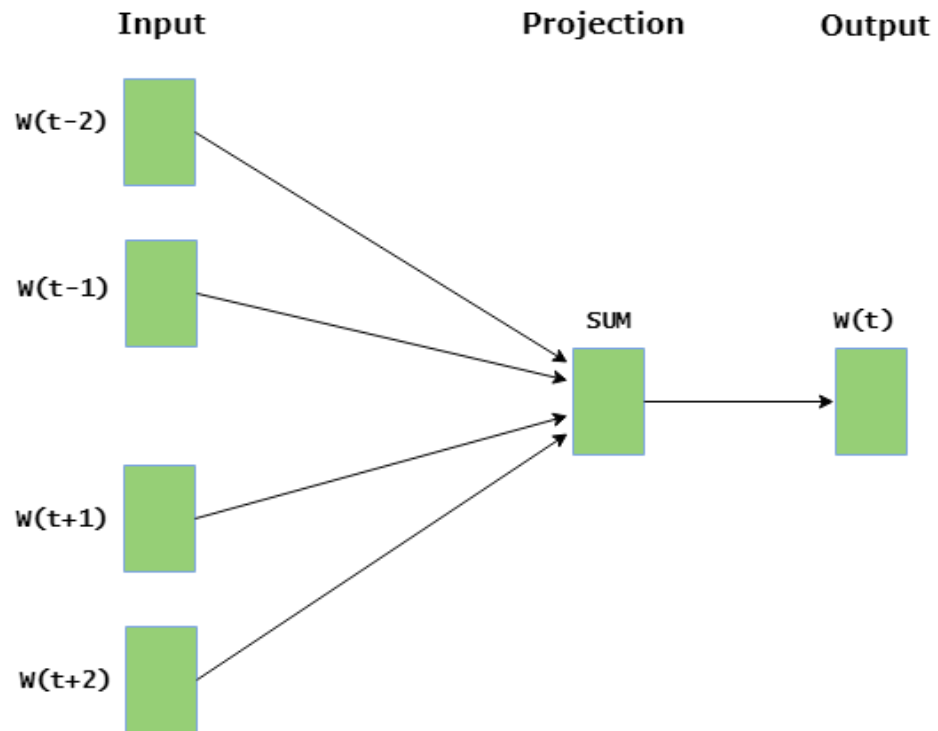


FIGURE 12. CBOW WORD2VEC

- b) Skip Gram.** On the basis of a single word provided at the input, multiple context words are generated on the basis of distances at the output. Hidden layer shows the dimension in which input word can be presented. Figure 13 shows the workflow of Skip Gram Word2Vec.

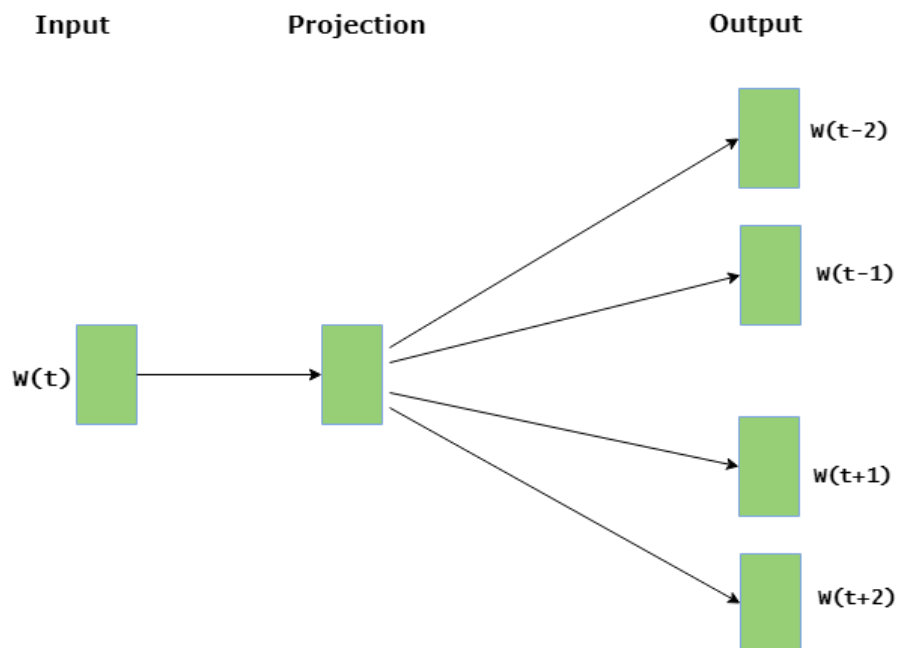


FIGURE 13. SKIP GRAM WORD2VEC

### A MACHINE LEARNING BASED METHODOLOGY

#### 5.1 Introduction

The challenge of Big Forensic Data (BFD) is the main focus of this research. The incredible amount of registry keys and values are needed to be processed through a system which automatically filters out the required forensically sound artifacts. So, in order to simplify registry forensics, a procedure is defined in this chapter to automatically produce relevant windows registry artifacts by utilizing Machine Learning's word embedding algorithm. Word2Vec is used in the proposed methodology for word suggestions on the basis distance finding. The details of methodology is discussed in the subsequent paragraphs.

#### 5.2 Proposed Methodology

On the basis of windows users' activities, registry is preserving the valuable users' relevant data. Such relevant data can be used to track the activities of a windows user; thus registry data helps in attribution. Most of the user perform predictable routine operations on a windows system i.e. working on MS Office, watching picture and videos, installation and uninstallation of different applications etc. A phase wise methodology as shown in Figure 14 is implemented in this research which takes registry data as input and automatically produces relevant attributable artifacts after filtration process.

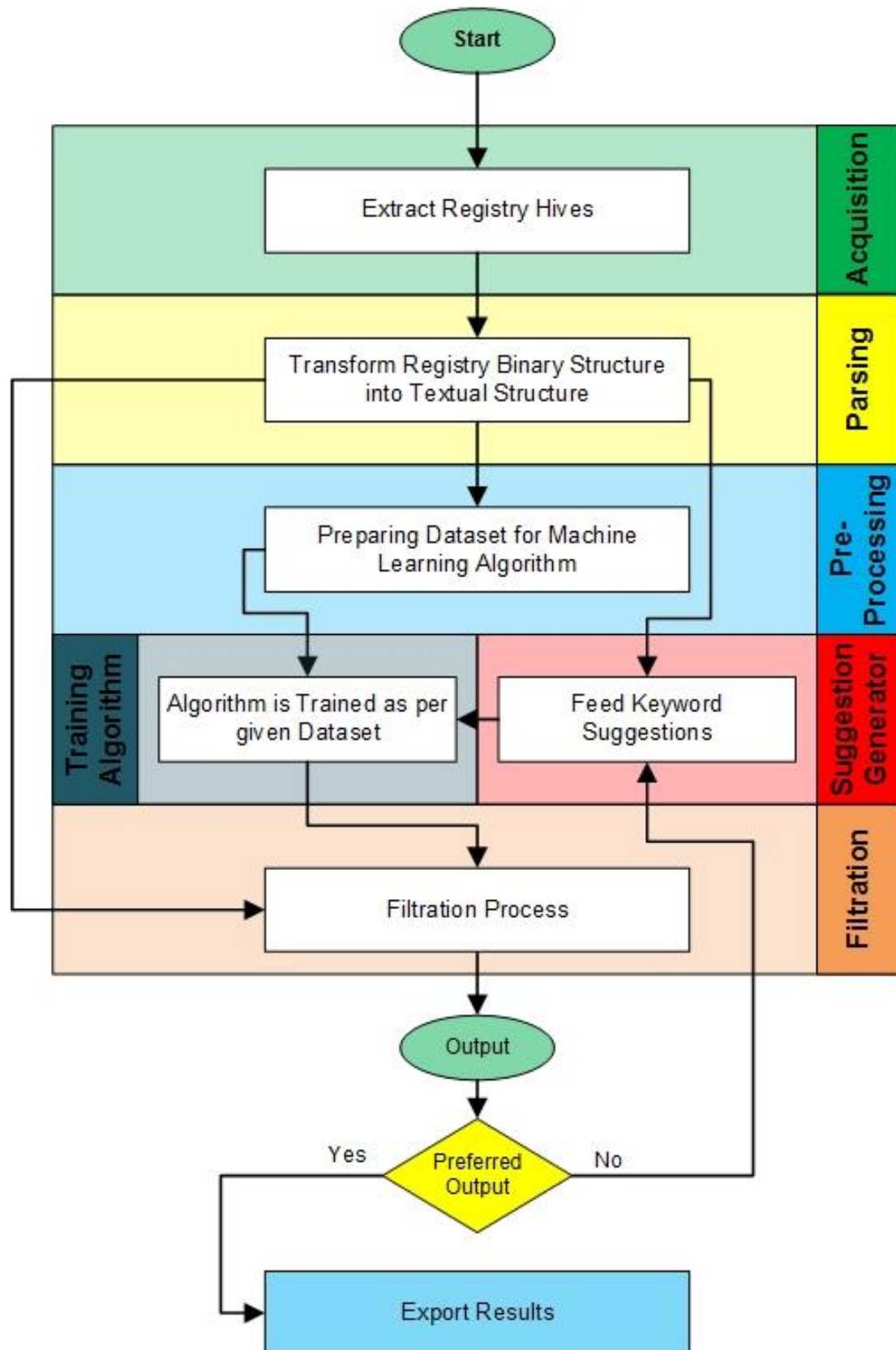


FIGURE 14. METHODOLOGY WORK FLOW

The proposed methodology consists of six phases and each phase is described as follows:-

### 5.2.1 Acquisition of Registry Hives

Methodology starts with the acquisition of Registry Hives as shown in Figure 15. Windows do not allow to access certain registry locations while windows is running. To keep original registry files intact and perform analysis in restricted areas, a registry image is taken using FTK Imager, a well reputed imaging tool. FTK Imager provides us with the exact replica of registry hive that is processed later on.

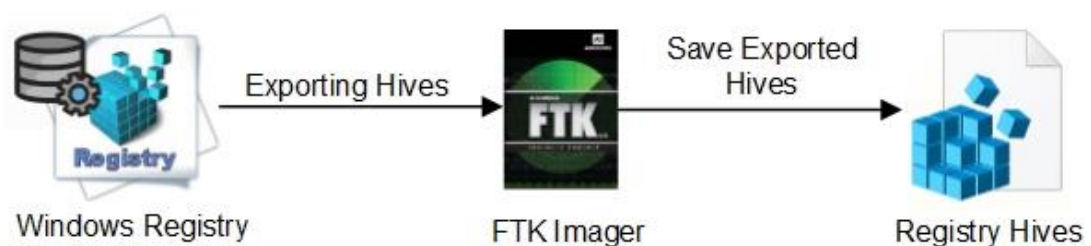


FIGURE 15. ACQUISITION PHASE

### 5.2.2 Parsing Registry Hives

Exported registry hives from phase one are in binary form. In order to apply word embedding algorithm, binary hives needed to be converted in textual format. In this phase registry hives are provided to an indigenously developed parser. Parser transforms registry hives native file structure into textual representation (JSON in our case) as output of this phase as depicted in Figure 16.

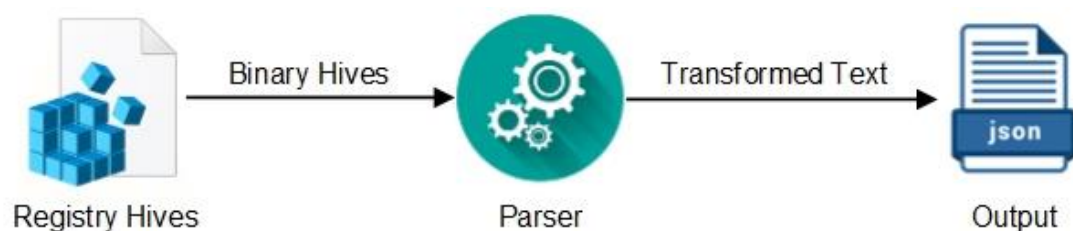


FIGURE 16. PARSING PHASE



### 5.2.3 Data Pre-Processing

Output JSON file from previous step is then provided to a customized Data Pre-Processor as shown in Figure 17. As JSON is a tree structured database and word embedding algorithm used in our case performs well on flattened data structure. Data Pre-Processing is done to flatten the data structure so that resultant dataset can be fed to the word embedding algorithm.

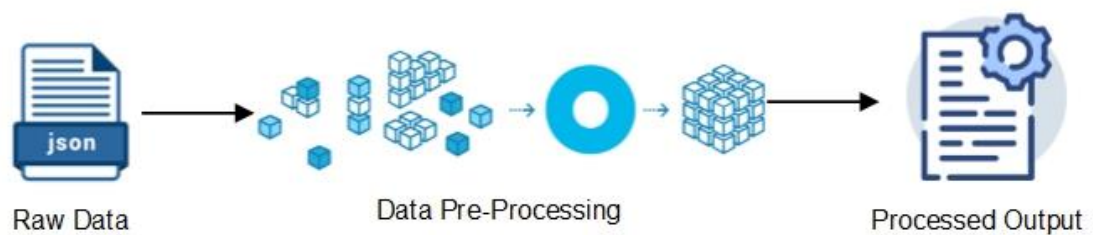


FIGURE 17. PRE-PROCESSING PHASE

### 5.2.4 Training ML Algorithm

Dataset generated in the previous phase is now given as input to the ML training process as shown in Figure 18. Word2Vec is chosen in this research to perform word embedding operations. It was necessary to find the distances of a particular word within the available registry textual data. Word2Vec helps in finding the relations of a word on the basis of distances. This process finds the relations and trains the model to make it ready to figure out relationships based on investigator's feedback.

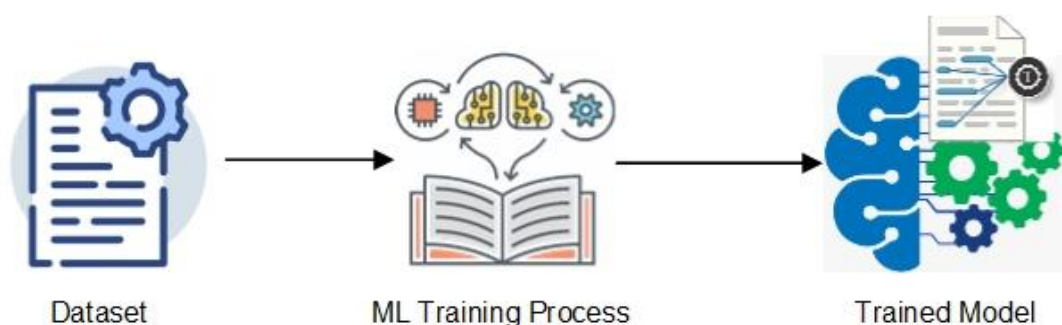


FIGURE 18. ML TRAINING PHASE

### 5.2.5 Relations based Keyword Suggestions

In this phase keyword generator is used on Parsed Hives from phase 2 to generate keywords on the basis of most frequently used words and also the applications available under the software registry key, as shown in Figure 19. Generated keywords then supplied to the trained model to find out the suggested relations on the basis of generated keywords. The resultant suggested relations are constructed with the help of distances. It is important to mention that suggested relations are dependent upon the amount of data i.e. more the data more accurate will be the results.

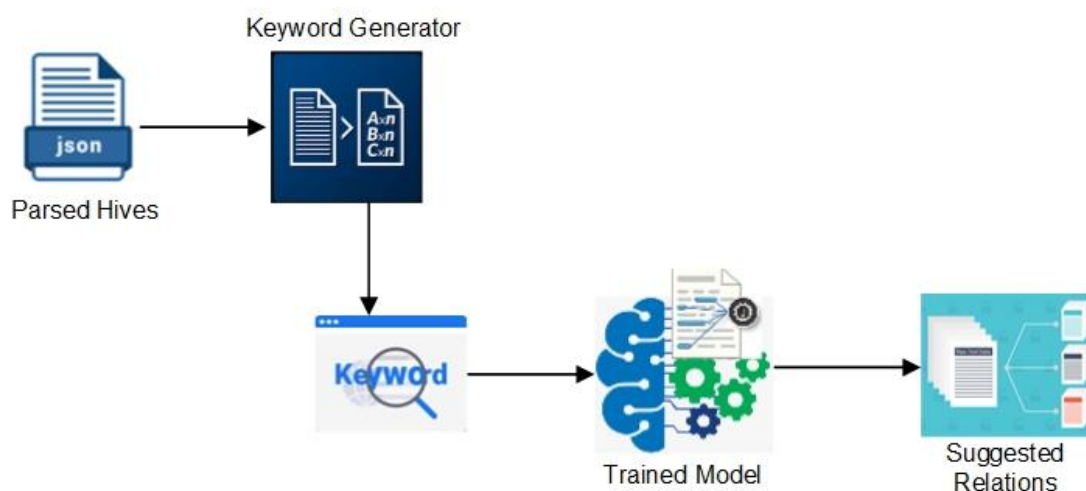


FIGURE 19. KEYWORD SUGGESTIONS PHASE

### 5.2.6 Filtration

In the final phase we used RegEx for filtration of data. The Suggested Relations (from previous step) and Parsed registry hives (from Phase 2) are provided to RegEx as input. On the basis of Suggested Relations RegEx filters the Parsed registry hives and gives the required filtered results as shown in Figure 20.

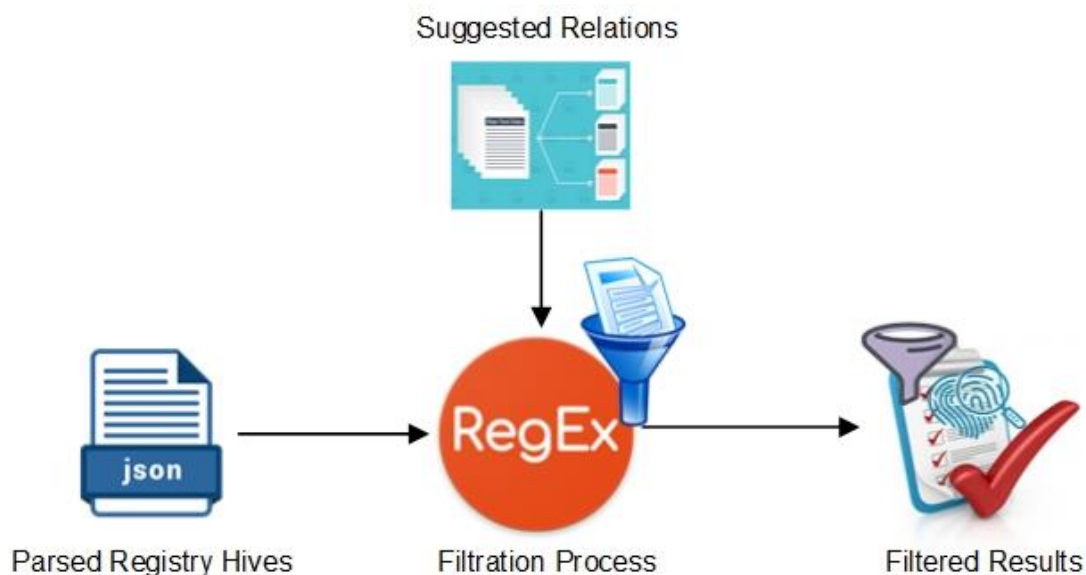


FIGURE 20. FILTRATION PHASE

All the above described phases are connected with each other. Figure 21 shows connections during different steps after combining all the phases in a single diagram. The methodology is an important step towards Windows registry forensics and will be very helpful in digital investigations involving Windows registry along with its huge frightening volume. The methodology can be applied to find out the relations of not only installed applications but also relations of any possible word which can be important in digital investigation of a digital crime case. Filtered results can be exported in excel format and are discussed in details in the next chapter.

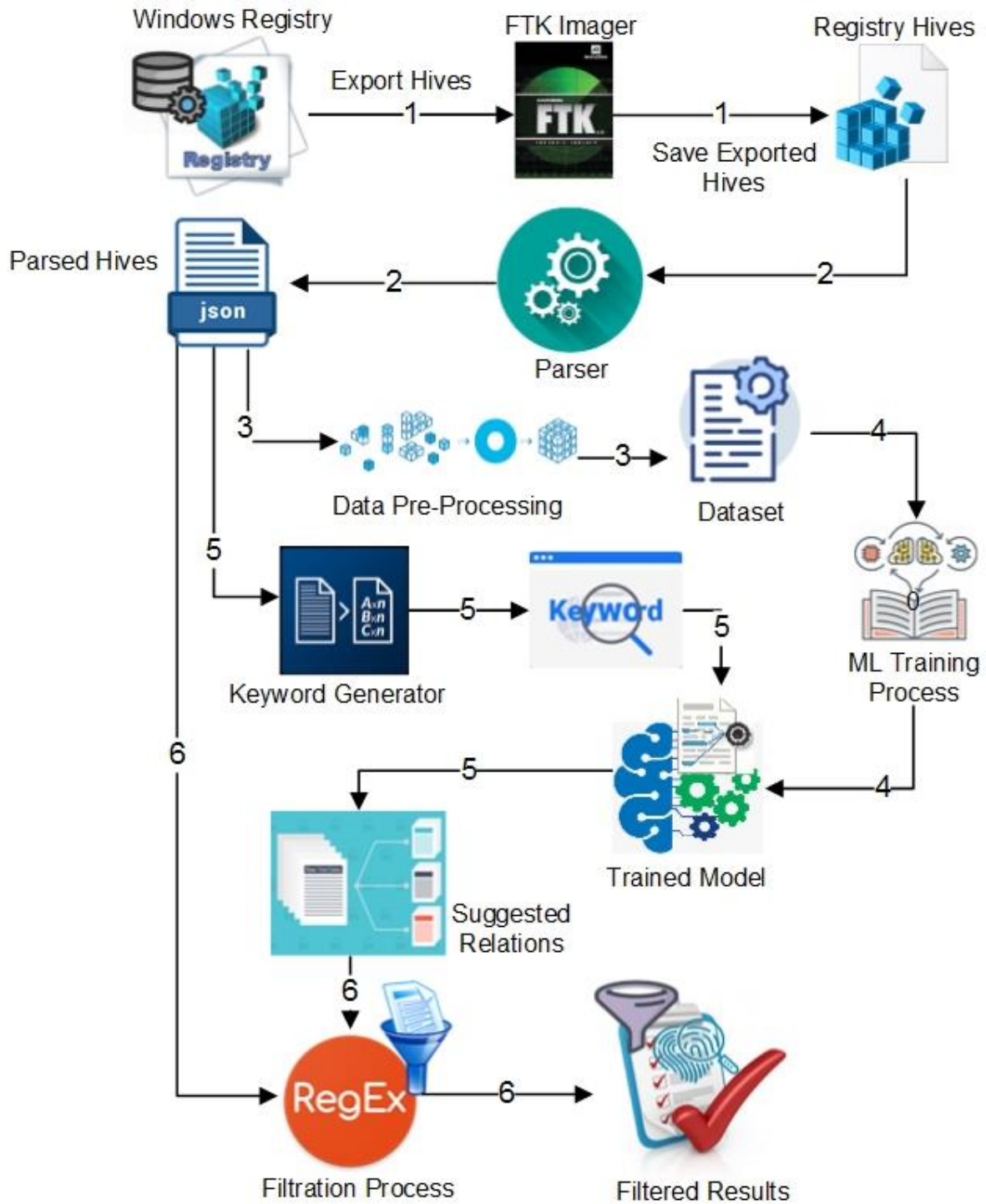


FIGURE 21. FUNCTIONALITY DIAGRAM - ALL PHASES COMBINED

### RESULTS

#### 6.1 Introduction

The defined methodology is applicable on any version of Microsoft Windows because the technique used will take the registry binaries and convert it into text. Furthermore, rest of the operations are performed on textual format of registry data which has no dependencies on the version of Microsoft Windows. Since Windows registry file format has different version i.e. Registry v4 (Win 9x, Win NT 4.0) and v5 (Win2000 till latest Win10), so this approach can be applied on both registry version. In this study, Microsoft Windows 7 Professional Operating System registry is extracted as use case and explored over the defined methodology.

#### 6.2 Experiments & Evaluation

Experiments were performed to figure out the most suitable parameters for *Word2Vec* algorithm. Parameters were chosen and evaluated on the basis of incremental changes as shown in Table 12. Considering the significance of digital evidence, one may understand that how important it is to not even lose a single evidence. The best way to avoid such expensive loss is to reveal all possible evidence relating to the case under inspection. Thus, in our scenario, *recall* is considered as the most important metric because it depicts the ratio of resulted potential evidence over ignored potential evidence. Higher percentages of *recall* means that most of the potential evidence are revealed which yields in interesting forensic artifacts. On the other hand, if the results contain more false positives than parameters with high *precision* values can be used to get optimal results.

TABLE 12. EVALUATION RESULTS

Parameters			Evaluation Metrics		
<i>Dim</i>	<i>Win Size</i>	<i>Min Count</i>	<i>Precision</i>	<i>Recall</i>	<i>F1 Score</i>
50	2	1	23.13	23.20	34.50
		2	27.50	28.81	41.14
		4	28.00	28.66	41.71
		8	31.33	32.74	45.87
		16	30.00	30.74	43.46
	3	1	25.62	26.87	37.76
		2	33.12	36.25	46.57
		4	32.00	32.81	46.94
		8	34.66	36.14	49.72
		16	36.66	37.48	50.74
	5	1	44.37	49.02	59.30
		2	41.87	48.47	56.58
		4	50.66	52.37	65.64
		8	44.66	46.22	59.97
		16	46.66	47.55	60.20
	10	1	59.37	<b>67.08</b>	70.81
		2	60.62	<b>67.15</b>	71.83
		4	<b>65.33</b>	<b>67.11</b>	<b>77.14</b>
		8	60.00	61.62	73.05
		16	54.66	55.55	66.21
15	1	54.99	60.83	66.79	
	2	52.49	58.95	65.54	
	4	58.00	59.55	70.93	
	8	58.00	59.70	71.74	
	16	49.99	50.88	63.01	
100	2	1	25.62	27.01	38.77
		2	26.25	30.00	39.30
		4	31.33	32.07	44.83
		8	32.00	33.55	46.79
		16	30.66	31.40	43.92
	3	1	31.84	33.33	46.28
		2	35.62	42.01	49.48
		4	37.33	38.74	52.34
		8	34.66	36.14	49.96
		16	35.33	36.14	49.70
	5	1	46.25	52.84	60.18
		2	45.00	49.65	59.41
		4	50.66	52.22	65.35
		8	52.00	53.70	66.73
		16	47.33	48.22	60.59
	10	1	56.87	64.02	69.25
		2	61.25	<b>68.40</b>	71.71
		4	58.66	60.29	71.66
		8	<b>64.66</b>	<b>66.29</b>	<b>76.51</b>
		16	53.99	54.88	65.83

	15	1	50.62	57.70	63.72
		2	58.12	65.90	70.30
		4	62.00	63.70	74.72
		8	58.00	59.62	71.51
		16	49.99	50.88	62.65
150	2	1	26.24	30.06	39.03
		2	26.25	32.50	38.15
		4	32.00	33.40	46.48
		8	31.33	32.74	45.88
		16	30.00	30.74	43.15
	3	1	30.62	31.45	44.85
		2	30.62	34.58	45.24
		4	36.00	37.62	51.37
		8	41.33	42.88	57.10
		16	33.99	34.74	47.03
	5	1	37.50	42.01	52.15
		2	47.50	52.15	61.40
		4	45.99	47.55	60.82
		8	48.00	49.62	63.17
		16	39.33	40.14	53.28
	10	1	59.37	65.76	70.44
		2	61.24	<b>69.02</b>	72.71
		4	59.33	60.96	72.86
		8	60.66	62.22	73.01
		16	52.66	53.55	65.01
15	1	48.75	54.58	61.89	
	2	60.00	<b>67.70</b>	71.63	
	4	55.33	56.96	69.02	
	8	56.00	57.55	69.95	
	16	49.33	50.22	62.34	
200	2	1	23.75	24.37	35.83
		2	26.24	29.37	39.36
		4	28.00	29.48	42.31
		8	32.66	34.14	47.52
		16	29.33	30.14	43.14
	3	1	31.25	32.63	45.72
		2	32.50	38.40	46.40
		4	37.33	38.81	52.50
		8	40.66	42.37	56.61
		16	37.33	38.22	51.37
	5	1	42.50	48.40	56.48
		2	50.62	58.40	64.07
		4	48.66	50.29	63.64
		8	45.99	47.55	61.63
		16	44.66	45.55	58.56
	10	1	53.75	60.83	66.24
		2	<b>65.00</b>	<b>72.84</b>	<b>75.22</b>
		4	<b>63.33</b>	<b>64.96</b>	<b>75.70</b>

		8	<b>64.66</b>	<b>66.44</b>	<b>76.34</b>
		16	53.33	54.22	65.45
	15	1	53.74	60.20	66.71
		2	56.87	64.72	69.36
		4	58.00	59.55	71.47
		8	59.33	60.88	72.44
		16	49.33	50.22	62.21
<b>300</b>	2	1	25.00	26.31	38.05
		2	26.87	33.12	39.22
		4	30.00	31.48	44.41
		8	30.66	32.14	44.56
		16	30.66	31.48	44.13
	3	1	30.62	32.56	44.16
		2	32.50	38.88	46.34
		4	34.66	36.29	49.66
		8	38.66	40.14	53.96
		16	33.33	34.14	47.30
	5	1	44.37	48.33	58.65
		2	43.75	50.27	58.22
		4	49.33	51.03	63.89
		8	50.00	51.55	65.23
		16	42.66	43.55	56.80
	10	1	59.37	65.90	70.80
		2	<b>63.75</b>	<b>70.20</b>	73.85
		4	61.99	63.70	74.35
		8	61.33	62.96	74.07
		16	54.66	55.55	66.38
	15	1	51.87	57.70	64.67
		2	57.49	64.02	70.08
		4	58.66	60.29	71.90
		8	58.00	59.62	71.38
		16	49.33	50.29	62.44

As per the evaluation results, the most feasible parameters to obtain optimum values are  $dim = 200$ ,  $win\_size = 10$  and  $min\_count = 2$ . Furthermore, it yields the best values for both recall and precision. After fixing the optimum values, it is necessary to find out a value for number of predictions on which maximum value of *Recall* can be achieved. Figure shows the relation between values of *Recall* and *Precision* with respect to increasing number of predictions.



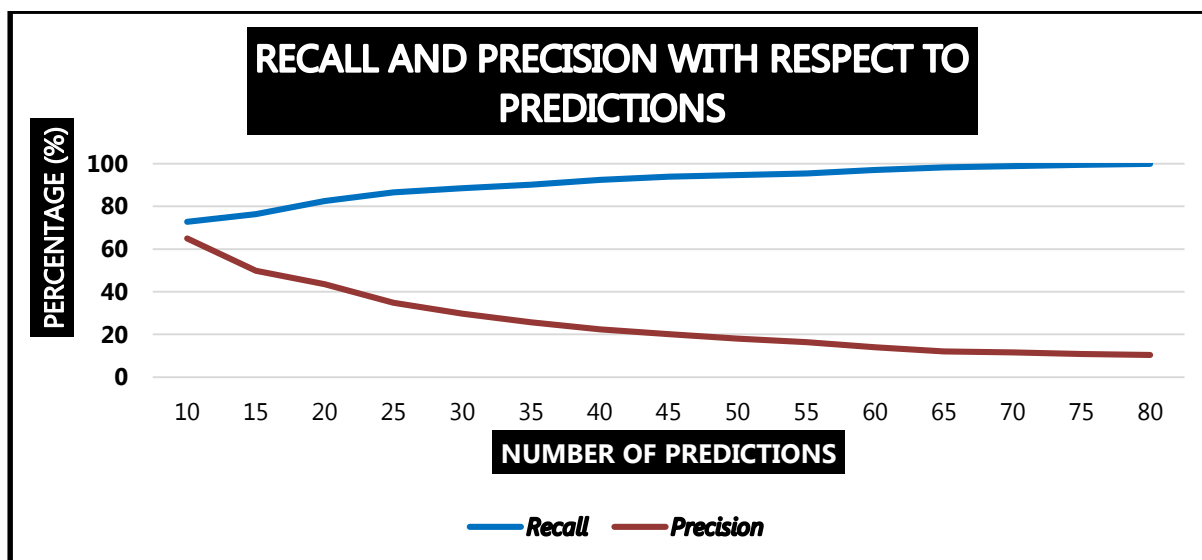


FIGURE 22. RELATION BETWEEN RECALL AND PRECISION

### 6.3 Results

Results are generated keeping in view the three different situations. First situation requires to extract the list of all the available applications whether installed or uninstalled. Second situation needs to find out all the Microsoft Word (if installed) documents name & locations from registry. Microsoft Word documents may be available on the resulted location or maybe not. Third situation desires all the relations of a particular user with the help of real name of that user to get valuable information about user's activities which will be helpful in attribution. Results of these situations are presented as follows:-

#### 6.2.1 Situation 1: List all Available Applications from Registry

For execution of the proposed solution, first step is to run the code as shown in Figure 22. On execution of code, JSON file is generated using the registry binary hive. JSON file is then transformed into flattened text file which is provided to Word2Vec algorithm for training the algorithm. Algorithm is trained automatically on the basis of generated flattened file.

```
(.venv) null@0x00:~/xCode/_apps/Regilizer$ ./main.py -i data/NTUSER.DAT -o output/out
[+] Generating JSON file
2020-01-17 00:48:12,431 : WARNING : consider setting layer size to a multiple of 4 for
2020-01-17 00:48:12,431 : INFO : collecting all words and their counts
2020-01-17 00:48:12,431 : INFO : PROGRESS: at sentence #0, processed 0 words, keeping
2020-01-17 00:48:12,455 : INFO : collected 10957 word types from a corpus of 156002 ra
2020-01-17 00:48:12,455 : INFO : Loading a fresh vocabulary
2020-01-17 00:48:12,468 : INFO : effective_min_count=2 retains 4825 unique words (44%
2020-01-17 00:48:12,468 : INFO : effective_min_count=2 leaves 149870 word corpus (96%
2020-01-17 00:48:12,486 : INFO : deleting the raw counts dictionary of 10957 items
2020-01-17 00:48:12,486 : INFO : sample=0.001 downsamples 67 most-common words
```

FIGURE 23. CODE EXECUTION

After the completion of algorithm's training process, list of available applications is displayed to the forensic investigator as suggestions based on application names as shown in Figure 23.

```
['3rd Eye Solutions', 'AccessData', 'Adobe', 'AKS-Labs', 'Analog Devices', 'Anvsoft', '
AppDataLow', 'ASPProtect', 'Avira', 'Bevywise', 'BitTorrent', 'BitTorrentPersist', 'bscd
esigner', 'Chromium', 'Clients', 'CnC Generals and Zero Hour', 'csastats', 'DMGR1.25',
'DMGR2.0.0', 'Freeware', 'FreshGames', 'GameHouse', 'GNU', 'GOG.com', 'Google', 'IM Pro
viders', 'INTEL', 'JEDI-VCL', 'LAV', 'Lavasoftware', 'Macromedia', 'Mendeley Ltd.', 'Micro
soft', 'MiniTool Software Limited', 'Mozilla', 'MPC-HC', 'Netscape', 'ODBC', 'Oracle', '
Policies', 'PuzzleLab', 'Python', 'QtProject', 'SimonTatham', 'SlavaSoft', 'Softland',
'undefined', 'VB and VBA Program Settings', 'VMware, Inc.', 'WinRAR', 'WinRAR SFX']
[+] Enter Keywords: word
2020-01-17 02:07:27,815 : INFO : precomputing L2-norms of word weight vectors
```

FIGURE 24. LIST OF AVAILABLE APPLICATIONS IN REGISTRY

### 6.2.2 Situation 2: Finding all Microsoft Word Documents from Registry

Next situation is to find all the Microsoft Word documents from given registry. For this purpose, forensic investigator will provide the keyword 'word' in the *Enter Keywords* input. In reply, Word2Vec will provide the all possible suggestions relevant to the 'word' as shown in Figure 24.

```
[+] Enter Keywords: word
2020-01-17 02:07:27,815 : INFO : precomputing L2-norms of word weight vectors
['powerpoint', 'excel', 'mru', 'locations', 'path', 'reading', 'excelname', 'word', '.d
oc', '.docx']
[+] Enter filter word: .doc
Total Records: 265
[['File MRU',
'\\Software\\Microsoft\\Office\\12.0\\Word\\File MRU',
'Item 1',
'[F00000000][T01D46141BE140AA0]*C:\\Users\\Administrator\\Downloads\\win reg '
'for.docx'],
('File MRU',
'\\Software\\Microsoft\\Office\\12.0\\Word\\File MRU',
'Item 2',
'[F00000000][T01D461415F436D90]*D:\\MSIS\\Semester1\\Computer '
'Security\\Research Project\\CS Semester Proj Task 3 4&5 - Amir Amin & '
'Farrukh Shabbir.docx'),
```

FIGURE 25. FINDING ALL MICROSOFT WORD DOCUMENTS FROM REGISTRY

Based on provided suggestions, investigator will provide *Enter filter word* to further narrow down search. *Enter filter word* option allows investigator to input multiple words separated with comma. These words can be entered irrespective of provided suggestions. Entered word(s) is then provided to RegEx engine. RegEx filters out the results from JSON file with respect to entered word(s). Results containing all available Microsoft Word documents are exported in CSV format as depicted in Figure 25.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	File MRU	\Software\Item 1	[F0000000][T01D46141BE140AA0]*C:\Users\Administrator\Downloads\win reg for.docx												
2	File MRU	\Software\Item 2	[F00000000][T01D461415F436D90]*D:\MSIS\Semester1\Computer Security\Research Project\CS Semester Proj Task 3 4&5 -												
3	File MRU	\Software\Item 3	[F00000000][T01D46140B309C470]*D:\MSIS\Semester1\Computer Security\Research Project\Final paper -presentation\My F												
4	File MRU	\Software\Item 4	[F00000000][T01D4613D8A76D550]*D:\MSIS\Semester1\Computer Security\Research Project\CS Term Proj - Amir Amin & F												
5	File MRU	\Software\Item 5	[F00000000][T01D45FBCBEBB350]*C:\Users\Administrator\Downloads\Pakistan Lecture Series 2018.docx												
6	File MRU	\Software\Item 6	[F00000000][T01D45D7FF0D2E5D0]*D:\MSIS\Semester1\Network Security\Assignment 1\Assignment1 - Amir Amin MSIS18 :												
7	File MRU	\Software\Item 7	[F00000000][T01D45C200BD16C50]*D:\MSIS\Semester1\Advance Crypto\Assignment No 1.doc												
8	File MRU	\Software\Item 8	[F00000000][T01D45C19D32EBB60]*D:\MSIS\Semester1\Network Security\Assignment 1\Task2\Assignment1 Task2 - Amir A												
9	File MRU	\Software\Item 9	[F00000000][T01D45641AFD91710]*C:\Users\Administrator\Downloads\RM Project Guidelines Dr Imran Mahmood.docx												
10	File MRU	\Software\Item 10	[F00000000][T01D45434B813B980]*C:\Users\Administrator\Downloads\CS Term Proj - Amir Amin & Farrukh Shabbir.docx												
11	File MRU	\Software\Item 11	[F00000000][T01D454347B3991B0]*C:\Users\Administrator\Downloads\Untitled 2.docx												
12	File MRU	\Software\Item 12	[F00000000][T01D452930D431FC0]*D:\MSIS\Semester1\Network Security\Assignment 1\Task1\Assignment1-Amir Amin MS												
13	File MRU	\Software\Item 13	[F00000000][T01D45292D25C3C70]*D:\MSIS\Semester1\Network Security\Assignment 1\Task1\INT_CCPR_CSS_PAK_27604												
14	File MRU	\Software\Item 14	[F00000000][T01D4527A644C3270]*D:\MSIS\Semester1\Advance Crypto\IS-842 Advanced Cryptography-1.docx												
15	File MRU	\Software\Item 15	[F00000000][T01D4512E52F4AFE0]*C:\Users\Administrator\Downloads\tyrian2000\Documentation\HELPME.DOC												

FIGURE 26. EXPORTED RESULTS OF MICROSOFT WORD DOCUMENTS

### 6.2.3 Situation 3: Discover Relations on the basis of Real Name of User

In this situation an investigator will be asked to find out the relevant Registry entries against a real name of some person. In our testing data ‘*Amir*’ is the real name of the computer owner. Investigator will enter the real name ‘*amir*’ in the *Enter Keywords* input. As a result suggested relations are displayed as shown in Figure 26. From suggested results investigator will pick a suggestion of his choice, in our case ‘*farrukh, amir*’ is entered to find out the relation between *Amir* and *Farrukh*. A total of 219 records were found. Figure 27 depicts the results of generated by the query.

```
[+] Enter Keywords: amir
2020-01-17 02:10:25,147 : INFO : precomputing L2-norms of word weight vectors
['farrukh', 'sl', 'amiramin', 'courses', 'datetime', 'jc', 'position', 'residence', 'sc
', 'shabbir', 'amir']
[+] Enter filter word: farrukh,amir
Total Reords: 219
(['Recent File List',
 '\\Software\\AccessData\\Registry Viewer\\Recent File List',
 'File2',
 'D:\\MSIS\\Semester2\\CF\\Paper\\Results\\Office2013 Results\\Registry DAT '
 'Files\\AmirUser\\NTUSER.DAT'),
```

FIGURE 27. REAL NAME BASED QUERYING

```
'Presentation - AmirAmin FarrukhShabbir Muddasir Waheed Zohaib Khan '
'(1).pptx'),
'Place MRU',
 '\\Software\\Microsoft\\Office\\15.0\\PowerPoint\\Place MRU',
 'Item 1',
 '[F00000000][T01D5C65A33057690][000000000]*D:\\MSIS\\Semester3\\Thesis\\Farrukh\\')
'Place MRU',
 '\\Software\\Microsoft\\Office\\15.0\\PowerPoint\\Place MRU',
 'Item 2',
 '[F00000000][T01D5C63C78DE0010][000000000]*D:\\MSIS\\Semester3\\Thesis\\Amir\\'),
'Place MRU',
```

FIGURE 28. REAL NAME BASED QUERY RESULTS

### CONCLUSION, LIMITATIONS AND FUTURE WORK

#### 7.1 Conclusion

Enormous amount of registry entries make it very difficult for an investigator to find out the forensically sound artifacts. While analyzing this much volume of data investigator may overlook some important artifacts which may result in severe consequences. Current registry forensic solutions does not have the capability to adapt to the registry changes made by a newly introduced application. In this research, the proposed solution is able to adapt to the changes made by a newly launched application. A tool is developed based on Machine Learning techniques to automatically collect forensically sound artifacts. The proposed methodology is robust enough to tackle the increased amount of registry keys and values introduced with the evolution of Windows operating system.

#### 7.2 Limitations

Although the solution has overcome the limitation of analyzing unknown applications and ever growing registry, but due to the lack of availability of pre-trained models on registry datasets and uniqueness of registry vocabulary as compared to English language literature, training the ML algorithm is a cumbersome task, hence not giving purely accurate results. Other limitations are the word suggestion produced by Word2Vec are prone to variations; thus may lead to spoiled results in certain instances. Spoiled results are not desirable in Digital Forensics.

#### 7.3 Future Work

Keeping in view the importance of Windows Registry Forensics, a vocabulary of words can be made available in future for training and testing the Machine Learning algorithms. It will not only help the digital investigations but also will be cope up with the windows registry evolution. In future, algorithms other than Word2Vec are to be tested to carry out the Windows Registry Forensics. Results of different algorithms will

be compared to find out the most precise, accurate and rigorous algorithm for the said purpose. Comparison of different algorithms will reduce the chances of erroneous results and will help the digital investigator in selecting the best ML algorithm for performing Windows Registry Forensic Analysis.

---

## REFERENCES

- [1] P. Čisar and S. M. Čisar, "General Directions of Development in Digital Forensics," *Acta Tech. Corviniensis - Bull. Eng.*, vol. 5, no. 2, pp. 87–92, 2012.
- [2] I. O. D. Chris, and D. David, "A New Approach of Digital Forensic Model for Digital Forensic Investigation," *Int. J. Adv. Comput. Sci. Appl.*, vol. 2, no. 12, pp. 175–178, 2011.
- [3] AT&T, "No Title," *financeonline.com*. [Online]. Available: <https://financesonline.com/cybercrime-statistics/>.
- [4] Netmarketshare, "Operating Systems Market Share." [Online]. Available: <https://netmarketshare.com/operating-system-market-share.aspx>.
- [5] H. Carvey, "The Windows Registry as a forensic resource," *Digit. Investig.*, vol. 2, no. 3, pp. 201–205, 2005.
- [6] S. B. Lee, J. Bang, K. S. Lim, J. Kim, and S. Lee, "A stepwise methodology for tracing computer usage," *NCM 2009 - 5th Int. Jt. Conf. INC, IMS, IDC*, pp. 1852–1857, 2009.
- [7] K. S. Lim, S. B. Lee, and S. Lee, "Applying a stepwise forensic approach to incident response and computer usage analysis," *Proc. 2009 2nd Int. Conf. Comput. Sci. Its Appl. CSA 2009*, no. December 2009, 2009.
- [8] T. D. Morgan, "Recovering deleted data from the Windows registry," *DFRWS 2008 Annu. Conf.*, vol. 5, pp. 33–41, 2008.
- [9] A. Singh, H. S. Venter, and A. R. Ikuesan, "Windows registry harnesser for incident response and digital forensic analysis," *Aust. J. Forensic Sci.*, vol. 00, no. 00, pp. 1–17, 2018.
- [10] A. Amin, F. Shabbir, S. Saleem, M. Waheed, and Z. Khan, "Microsoft Word Forensic Artifacts in Windows 10 Registry," in *2019 International Conference on Applied and Engineering Mathematics, ICAEM 2019 - Proceedings*, 2019.
- [11] L. Bruno, "No Title No Title," *Journal of Chemical Information and Modeling*, 2019. [Online]. Available: <https://support.microsoft.com/en->

- us/help/256986/windows-registry-information-for-advanced-users.
- [12] Microsoft, "No Title." [Online]. Available: <https://docs.microsoft.com/en-us/windows/win32/sysinfo/registry>.
- [13] ComputerHope, "No Title." [Online]. Available: <https://www.computerhope.com/jargon/r/registry.htm>.
- [14] D. Bem, F. Feld, E. Huebner, and O. Bem, "Journal of Information Science and Technology www," 2008.
- [15] P. Čisar and S. M. Čisar, "Methodological frameworks of digital forensics," *SISY 2011 - 9th Int. Symp. Intell. Syst. Informatics, Proc.*, pp. 343–347, 2011.
- [16] S. Al-Fedaghi and B. Al-Babtain, "Modeling the forensics process," *Int. J. Secur. its Appl.*, vol. 6, no. 4, pp. 97–108, 2012.
- [17] D. Hintea, R. Bird, and M. Green, "An investigation into the forensic implications of the Windows 10 operating system: Recoverable artefacts and significant changes from Windows 8.1," *Int. J. Environ. Sustain. Dev.*, vol. 16, no. 4, pp. 326–345, 2017.
- [18] R. M. Saidi, S. A. Ahmad, N. M. Noor, and R. Yunos, "Windows registry analysis for forensic investigation," in *2013 The International Conference on Technological Advances in Electrical, Electronics and Computer Engineering, TAECE 2013*, 2013.
- [19] M. N. Faiz and W. A. Prabowo, "Comparison of Acquisition Software for Digital Forensics Purposes," *Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control*, vol. 4, no. 1, p. 37, 2018.
- [20] A. Arshad, W. Iqbal, and H. Abbas, "USB Storage Device Forensics for Windows 10," *J. Forensic Sci.*, vol. 63, no. 3, pp. 856–867, 2018.
- [21] H. Binjuraid and M. Mat Din, "Case Based Interpretation of Windows 10 Registry Forensics," *Int. J. Innov. Comput.*, vol. 8, no. 1, pp. 43–47, 2018.
- [22] E. Wahyudi, I. Riadi, and Y. Prayudi, "Virtual Machine Forensic Analysis And Recovery Method For Recovery And Analysis Digital Evidence," *nternational J. Comput. Sci. Inf. Secur.*, vol. 16, no. 2, pp. 1–7, 2018.



- 
- [23] R. M. A. Mohammad and M. Alqahtani, "A comparison of machine learning techniques for file system forensics analysis," *J. Inf. Secur. Appl.*, vol. 46, no. September 2016, pp. 53–61, 2019.
- [24] E. I. Edem, C. Benzaid, A. Al-Nemrat, and P. Watters, "Analysis of malware behaviour: Using data mining clustering techniques to support forensics investigation," *Proc. - 5th Cybercrime Trust. Comput. Conf. CTC 2014*, pp. 54–63, 2015.
- [25] M. N. A. Khan, C. R. Chatwin, and R. C. D. Young, "A framework for post-event timeline reconstruction using neural networks," *Digit. Investig.*, vol. 4, no. 3–4, pp. 146–157, 2007.
- [26] D. Ocean, "Machine Learning Intro." [Online]. Available: <https://www.digitalocean.com/community/tutorials/an-introduction-to-machine-learning>.
- [27] Analyticsvidhya, "No Title." [Online]. Available: <https://www.analyticsvidhya.com/blog/2019/07/ultimate-list-popular-machine-learning-use-cases/>.
- [28] R. Montasari and R. Hill, "Next-Generation Digital Forensics: Challenges and Future Paradigms," *Proc. 12th Int. Conf. Glob. Secur. Saf. Sustain. ICGS3 2019*, pp. 205–212, 2019.