Volume 8, No. 8, September-October 2017



**RESEARCH PAPER** 

Available Online at www.ijarcs.info

# MACHINE LEARNING FORENSICS: A NEW BRANCH OF DIGITAL FORENSICS

Prerak Bhatt

IFS, Gujarat Forensic Sciences University, Gandhinagar, Gujarat Parag H. Rughani, Ph. D. IFS, Gujarat Forensic Sciences University, Gandhinagar, Gujarat

*Abstract*— The objective of this research is to understand how machine learning can be used in digital crime and its forensic importance, setting up an environment to train artificial neural networks and investigate as well as analyze them to find artefacts that can be helpful in forensic investigation.

Keywords- Machine Learning, ML forensics, AI forensics, TensorFlow, AI related crimes

# I. INTRODUCTION

Artificial Intelligence (AI) and Machine Learning (ML) has been around since a long time but it is now that we have enough computational power to effectively develop strong artificial neural networks (ANN) in a sensible time frame with the help of strong hardware and software support.

Companies like Google, Amazon, Samsung etc. are heavily investing in AI technologies and funding the research. Google CEO Sundar Pichai announced the company's vision to be "AI-First" at Google I/O 2017 and quoted "It's all about a transition, from searching and organizing the world's information to AI and machine learning." [1].

Google also unveiled "TensorFlow Research Cloud" program which will provide researchers with access to 1000 cloud TPUs (Tensor Processing Units) for free with a condition to open source their code and research. [2]

Since AI is becoming widely available to more and more people, the potential of the technology to be used for malicious purposes also increases significantly. To counter this and be prepared for forensic challenges regarding crimes committed with AI or ML, forensic evaluation and analysis of the technology is necessary.

This paper demonstrates implementation of a machine learning open source program "DeepQA" and forensic analysis of the same while the program was in training and testing modes. This paper also lists out some important artifacts findings that can be taken as a reference for cases in future to prove or determine that a machine learning technique based on TensorFlow was used on provided evidence.

# II. FORENSIC IMPORTANCE

AI and ML has great advantages and holds a bright future ahead. But the same technology can inevitably be used to craft, automate and execute some serious crimes that can also be deadly for people.

For instance, hackers can develop an ANN that scans new versions of popular apps for unknown vulnerabilities, exploit them and/or report the vulnerability to the hacker. If this process is done manually, it can take long time to pentest an app. But with the help of ML it can be done really quickly and can be done on multiple different apps at the same time with

machine efficiency. It makes the job for hacker really easy and quick.

Hackers are available to rent on dark web. It is a possibility now that AI powered bots will replace them and do the job more efficiently and quickly than a human being can. Earning more money to hacker than he did before.

The task in 2016 DARPA (Defense Advanced Research Projects Agency) Cyber Grand Challenge was to create an AI that can correct the provided buggy code itself, patch vulnerabilities present in its own system and look for intrusions by opponents with minimal human interaction. The winning prize was \$2million. The competition lasted for about 8 hours. [3]

"Spear phishing is going to become really, really good when machine learning is incorporated into it on the attacking side," says Dave Palmer, director of technology at Darktrace, a cybersecurity firm which deploys machine learning in its technology. [4]

So, there are so many possibilities where ML and AI can make a criminal, hackers or a terrorist's job easy and quicker.

And this is why, ML and AI holds great forensic importance. It is a new field to dive in for forensic investigators and the scope of research is really big. There is no research found on the algorithms or frameworks of ML that suggest how to investigate or identify if any AI or ML technology was used in commitment of a digital crime.

# **III. Environment setup**

We need a specific environment setup in order to develop ANN based programs. It requires a powerful CPU and/or GPU because training of an ANN model is a resource consuming task.

# A. Hardware

- **CPU**: Intel Core i5 6600k
- GPU: Nvidia GeForce GTX 1070
- **RAM**: 8GB
- B. Software
  - **OS**: Ubuntu 16.04

- Parallel Computing Platform: Nvidia CUDA
- ML Library/Framework: cuDNN & TensorFlow 1.0.0
- Language Platform: Python 2 & Python 3
- ML Program: DeepQA [5]
- Dataset: Cornell Movie Dialogs [6]

And other dependencies for above mentioned programs.

#### C. Why use GPU?

Training deep neural networks can be a time-consuming process. It involves a big amount of matrix multiplications and other mathematical operations that if parallelized, can boost up the calculation time significantly.

A single workstation CPU in current scenario might have 8-10 cores, while a single GPU can have thousands of cores. Although the GPU cores are slower than CPU cores, the large number of cores makes that redundant.

Nvidia GeForce GTX 1070 GPU has 1920 CUDA cores clocked at 1506MHz with 8GB of VRAM.

### D. Why TensorFlow?

Google open sourced TensorFlow on November 9, 2015. Since then, it is the most sophisticated and well written library for Machine Learning. It supports CPU and GPU acceleration.

Because of its open source nature, it has attracted a lot of machine learning programmers and they are using it in various creative ways to build different types of programs and services.

Now after the announcements at Google I/O, the use of TensorFlow will spread more between developers and more open source projects will be pushed out.

Because of these reasons, I have chosen TensorFlow library to build a ML ANN.

#### **IV. IMPLEMENTATION**

# A. Ubuntu 16.04 & nVidia utilities:

I installed Ubuntu 16.04 with a separate /home, / (root) and SWAP partitions on a workstation. Installed all necessary programs such as Python 2 and Python 3 and other dependencies that are required.

Installed latest nVidia graphics drivers, as well as installed cuda and cudNN utilities that provide tensorflow a bridge between Python and GPU for training a neural network.

# B. Tensorflow:

Cloned tensorflow from its GitHub repository.

Configured tensorflow installation by running ./configure script in tensorflow directory. Configuration included specification of location of cuda installation, cudnn installation, compute capability of your GPU, python installation directory etc.

After configuring tensorflow correctly, I built a pip python package and installed tensorflow as a plugin on my python 3 installation.

# C. DeepQA ChatBot Program:

DeepQA is an open source Neural Conversational Model. It uses a RNN (seq2seq model) for sentence predictions. It is based on Python and TensorFlow. The advantage of this program is that it supports a various number of conversational datasets available for research purpose.

DeepQA also provides code to setup a Django web server that gives the chatbot a nice graphical interface to play with.

The developer of this program is very responsive to queries and keeps the project up to date.

Because of this program's versatility and activeness, I chose this program to train an ANN chat bot.

#### D. Training chat bot:

To start training a neural network on "Cornell movie dialogs" dataset, I entered the following command:

\$python3 main.py --corpus cornell

889	Terminal	File	Edit \	/iew S	Search	Termina	l Help			
	ep 33300									
	ep 33400									
	ep 33500									
	ep 33600									
	ep 33700		Loss	1.32	P	erplexi				
Training						St. St. State		547/547	[01:41<00:00,	5.56it/s]
Epoch fi	nished i	n 0:	01:41	.0476	40					
	och 4/30			002)						
Shufflin										
	ep 33800									
St										
	ep 34000									
Checkpoi			savin	g mod	lel (	don't s	top the	run)		
Model sa										
St	ep 34100					erplexi				
St	ep 34200		Loss			erplexi	ty 3.88			
Training	: 100%							547/547	[01:43<00:00,	
Epoch fi	nished i			.7865	85					
Ep	och 5/30			002)						
Shufflin	g the da	itase								
St	ep 34300		Loss			erplexi	ty 3.25			
Training	: 8%							44/547	[00:08<01:33,	5.38it/s]

Figure 1: Training on

There are many different parameters and variables are being displayed in figure 1. Explanation of them is as following:

An epoch usually means one iteration over all of the training data. For instance, if you have 20,000 images and a batch size of 100 then the epoch should contain 20,000 / 100 = 200 steps.

The loss measure error between two tensors, or between a tensor and zero. These can be used for measuring accuracy of a network in a regression task or for regularization purposes.

Perplexity metric in ML is a way to capture the degree of 'uncertainty' a model has in predicting (assigning probabilities to) some text. It is related to Shannon's Entropy. Lower the entropy (uncertainty), lower the perplexity.

You can finish the training any time by pressing ctrl + c in the terminal. It will save the model as a .ckpt file in /save directory and exit the program.

You can resume training from the same step it left off later on if you want to.

Training an ANN generally takes a good amount of time depending on the size of provided dataset and how many epochs (one full training cycle) you are running your program for. Also the parameters you use in order to train the ANN effects the training time.

It took me about 7 to 8 hours to train an ANN model based on default parameters with 30 epochs and used "Cornell Movie Dialogs" for dataset. Initially, it did not give me good results.

But by trying and setting different values in parameters like learning rate (lr), max sentence length, etc. started to get me better results.

I spent more than 24 hours in total to train different models to get better results. Each try consisted of 30 epochs.

# E. Testing chat bot:

To test the trained model and see how well has it trained based on our dataset, I entered the following command:

# \$python3 main.py --test interactive

It provides a command line interface where you can type a question or a message and the chatbot will reply to it.



Figure 2 - testing the chatbot

The replies the bot is giving to questions in figure above. They are not really great but they are somewhat contextual based on the questions asked to it.

Training it on a better dataset and for longer timing with proper learning rate and other parameters can give you better results.

Now imagine we provide this model a dataset that consists of conversations between a support employee of bank and a client. If we train it properly and long enough then it will be able to successfully make the client believe that he is talking to a real legit person and he would trust him enough to reveal his information to him.

You can visualize the computational graph, the cost of the ANN and word embeddings for our model with TensorBoard, just run tensorboard --logdir save/ command.

Word embedding is the collective name for a set of language modelling and feature learning techniques in natural language processing (NLP) where words or phrases from the vocabulary are mapped to vectors of real numbers in a lowdimensional space relative to the vocabulary size. [7]

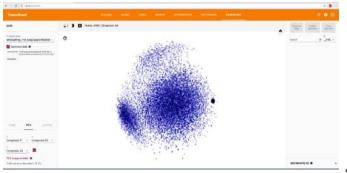


Figure 3 - TensorBoard word embeddings

The embeddings of our trained model can be seen in the screenshot above. It is pretty dense. Means it is a well-trained model containing a big amount of word vectors.

# V. FORENSIC ACQUISITION AND ANALYSIS

After training a functional neural network that can give out decent output, its time to forensically analyze the system to find artefacts that help us determine that the system was used in generation and testing of a neural network based on TensorFlow.

# A. Tools used:

- LiME: "Linux memory extractor" (LiME) is used to take live RAM dumps in .lime and .raw formats. [8]
- **Rufus**: To create a bootable Ubuntu 16.04 USB thumbdrive. [9]
- **Disks Utility:** It is a part of Ubuntu live system that lets you create images of different partitions or whole disk. [10]
- EnCase: EnCase is used to investigate disk images and RAM dumps to find relevant artefacts and to make a report based on findings and other technical information about the system. [11]

# B. Live memory capture with LiME:

LiME is a Loadable Kernel Module (LKM) which allows for volatile memory acquisition from Linux and Linux-based devices.

LiME utilizes the insmod command to load the module, passing required arguments for its execution.

After cloning the source code of LiME from GitHub, it is needed to make a kernel module compatible with your Linux kernel. You cannot load a kernel module that is made for another kernel on your kernel. It can be fatal in some cases for the OS.

I loaded the LiME kernel module in the kernel while the DeepQA program was in training mode.

O Terminal File Edit View Search Terminal Help psyntist@Psyntist:-/LiNE/src\$ sudo insmod lime-4.4.0-71-generic.ko "path=/home/psy ntist/besktop/TratningRAMdump.lime" format=raw [sudo] password for psyntist:

Figure 4 – LiME while training

After taking the RAM dump while program was in training mode, I put program in testing mode and again took a RAM dump following the same way.

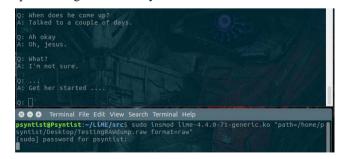


Figure 5 – LiME while testing

So now we have two different RAM dumps. One while the system was in training mode and one while the system was in testing mode.

# 1. TrainingRAMdump

#### 2. TestingRAMdump

#### C. Disk acquisition with 'Disks' Utility:

"Disks" is a tool that comes preinstalled with Ubuntu 16.04. It lets you manage your hard disk partitions. You can create new partitions, edit partitions, shrink, extend, mount, unmount and take logical images of partitions in .img format.

I created an Ubuntu 16.04 live bootable USB thumb drive and booted it up on my system.

I launched Disks utility and selected /home partition. Clicked on settings icon on left and selected 'create logical image' of the partition and provided the location to store a bitby-bit image of /home partition.



Figure 6 - /home image

I did the same procedure with the remaining partitions respectively, / (root) and swap.

So now we have logical images of all three partitions used on Ubuntu that will be loaded on EnCase for investigation.

The reason to acquire these partitions is explained below:

**/home**: DeepQA program is hosted on this partition as well as RAM dumps I took with LiME are stored here.

/ (**root**): Installation of TensorFlow and other dependency programs were done in this partition. Plus this directory is the parent of all directories on Ubuntu.

**SWAP**: Swap is used for paging. So it might have some volatile data stored that might be useful for the investigation.

## D. Forensic Analysis on EnCase:

Now comes the most interesting part of this project. Analyzing the RAM dumps and hard disk images to find relevant artefacts.

I chose EnCase to analyse the evidence because EnCase provides state of the art solutions for evidence analysing, processing and report generating. The interface is also easy to use and clean.

Biggest advantage to use EnCase is that it can be cited in court of law in USA, India and other major countries.

I created a new case on EnCase and entered appropriate information such as the name of case, case number, examiner namem case ID etc.

After creating the case, I added evidence files one by one. First off, I started with adding the logical image of /home partition. After adding the image of /home partition as an evidence file, it is time to acquire the same evidence. EnCase makes .E01 image of the raw image of the evidence we provided in acquiring phase.

After acquiring the evidence image, I put the acquired evidence image on processing. Selected appropriate processing options like System Info Parser, File Carver, Personal Information extractor, Linux Artefact Parser, etc.

I followed the same procedure for acquiring and processing for next two logical images, / (root) and SWAP.

Processing SWAP partition did not give any categorized data it was shown as unallocated space but, some RAW data can be found from that unallocated space.

After adding, acquiring and processing all evidence files EnCase Evidence window looks like this:

	O Very Lidence	E - G Coen G Trage  # Remove C Restan In Lodets Paths G Change	Cathes III	Date View	(/h Process Svidence + 9, + ) []	. 0
	de O timeine		a core na	Par. Tatr		
D -	JA + [] Selected 0/1					
	Name	Primary. Fach	Evidence Faths	Extra Pacha	6V/D	
014	D home partition	FoProject Analysis/Machinet.earringForensics/EvidenceCacheVhome partition/Ex01			5882010289848080808949446771668	- ExProject
011	C root leage	PhProject Analysis/MachineLearningForensics/EvidenceCachelococt Image.Ex03			47d13ec1d9546dc92371fdbe2ddb3c	4 http://
011	I Seto partice	PsProject Analysis/MachineLearningForensics/EvidenceCache/Gwap partition/Ex01			4972r627r0571cct8cta56a27e848d2	PhProject
4						
- fe	es 🖩 Report 🗉 Ca	nsy A bionoritaty Biossiters if Aquasoninti & Heathers & W	asing Sectors	A occ	egn ≣ 0 1 ⊥ =	C Look of
	es 🖩 Report 🗉 Ca	nov A biomorfethe 🕷 boartens 🖉 kouedon mb Å feedbron 🛦 Ve	asing Sectors	A occ	195 II # 1 1	

Figure 7 - All evidence images

# E. Findings:

I started the analysis of evidence and found some concrete artefacts explained below:

#### **Tensorflow Installation location:**

One of the most primary and important artefact is to find out if TensorFlow is installed on the system.

TensorFlow is a Python library. So, first check the location of Python installation and then look for tensorflow inside it.

On any Linux based OS programs are installed under /usr directory so it is the first directory one should consider to analyze for Python installation.

Tensorflow is installed under /usr/local/lib/python3.5/dist-packages/tensorflow directory.

1.01	hotestor Manager + A Evidence +								
	Ø view finities + ⊟ + □ Book		T Gotoffe * Tags + B	Review Package + S Rose	Search Search	a - 15 Tretes	· 2 1cm	er · (h Proces	B Deke - 1
	OCH PyAudio-0.2.10.egg-info *	100 1	able C Timeline C Gallery						
	COn pyttsk		= + B + D Selected 0/351	297					
	Color ma		Name		File Ext	Leg-cal Side		Stagory	Signature Analysis
	OCH speech, recognition	C 1	_pytade_			4,096	Folder		
	OF SpeechRecomption-3.6.0.e	02	e conerilo			4,096	Folder		
	CO Insoften	0.3	tore .			4.096	Folder		
	- D C in signatifita	0.4	🖿 examples			4,096	Folder		
	-O D core	0.5	Include			4,096	Folder		
	<ul> <li>D In examples</li> <li>D In indude</li> </ul>	0.6	E python			4,096	Folder		
	IDD python	07	tensorboard			4,096	Folder		
	ADD is tensorboard	01	tuois			4,096	Folder		
	<ul> <li>Dig tools</li> <li>Dig tools</li> </ul>	0.9	C _HLay		DV.	1,438	Note	-	
	1000 tests 1000 untemplate								
Telo	# Report I Ted I Hei I p		E Doc E Transprot R R	cture El Console 🖛 🖉	2.0	C Look iff	D Co	noton ¥ Fiter	1 EnGeript
Diges	ns 🤼 Codepage + A Text Style +	18 10	ut 🖩 Fed Next (0) * 🔹	E Compressed Vew	5. A		3 /	Edt New	# Delete 1
195 (D) 127 143 143		454	prélais II. pre					Lise?	

Figure 8 - TensorFlow location

Interestingly, this directory also contains some interesting Python libraries that can be used as a part of a ML program. Such as speech\_recognition, pyttsx, etc.

# Searching for keywords:

#### Seq2seq keyword:

seq2seq (sequence to sequence) is a class of tensorflow that is used in developing a sequence to sequence, RNN (recurrent neural network) model.

DeepQA program is based on seq2seq modelling and is a recurrent neural network. So the possibility to find this class used in creation of the model is high.

# Netflix keyword:

Word Netflix was a part of our dataset I used to train our ANN. So using this as a keyword to search to see if we can find it in RAM dumps or on SWAP partition.

#### GTX1070:

If training of ANN program was done with tensorflow and GPU, it will include the name of the GPU used in the training at least somewhere in volatile data or in parameters of tensorflow.

Added gtx1070 as a keyword to see if we find some artefacts related to it as I used gtx 1070 GPU to train the chatbot.

## Keyword hits:

It takes a good amount of time to analyze all evidence files for the provided keywords to EnCase. But it checks all evidence files thoroughly and even shows if keyword hit was found in unallocated space.

After the processing of searching for keywords finishes, EnCase shows you all the keyword hits in one window of Keywords. It shows the number of files and number of hits the keyword has got right next to the name of keywords. It can be seen in the screenshot below.

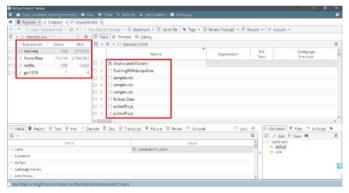


Figure 9 - Keyword hits

# Seq2seq keyword hit:

Surprisingly seq2seq keyword got 277,875 number of hits in all evidence images. Meaning it has been used a lot in 208 number of files. I analysed some of those files and found following results.

Found the python script file of DeepQA chatbot.py containing the seq2seq keyword. It can be seen that tensorflow class seq2seq class is used in the code of this file.

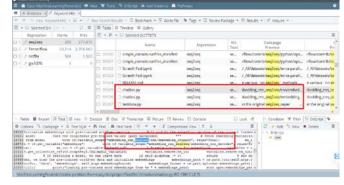


Figure 10 - seq2seq hit

The same keyword was also found in the compiled chatbot file chatbot.pyc it confirms that the script was indeed ran at least once on the system. The pyc (Python compiled) file only generates once the program executes at least once.

11	- Distence = P	Xeyword H	t5 H						
+-	4 O Venir Neys	entites -	田 =128	ivi Sewith Res	ults - 🗏 Bookmark = 🗄 Go to	fle   🗣 Tags = 🗇 Review P	ackage +	III Results +   - Acquire +	
Π.	- L1 Selected 0/4		C 1	III Table 4	D Timeine 🗐 Gallery				
	Expression	tt.ems	Nita	II + 2	<ul> <li>Selected 0/277875</li> </ul>				
	/ seq2seq	208	277,875		Nama	Depression	Hit	Codepage	17
3.2	P Tensorflow	20,214	2,708,553				Text		#74
11	2 netfix	506	1,522	CI 227711	🖸 index.md	sedged	58_	contribulegacy_seq2seq.md#seque_	. xeetblegacy_M
	P gtx1070	1	4	C) 277754	El index.md	sedjacá	18-	<ul> <li>Filed.mm.seq2seq"%L/L/apl.d.</li> </ul>	- "l'tied_rro_se
				0.227/215	C index.md	seq2seq	-98	contribulegacy_seq2seq.mdatled_rm	xontribilegacy_se
				C 277771A	🗋 index.md	perCper	58	militied man seq2seq) * **[Lines.	mditted_mn_se
				CI 2277717	C shattiot.cpython-35.pyc	seq2seq:	58	bedding_mm_seq2seq/msvembed	shoulding ron se
				CI 277718	C chatbol.cpython-35.pyt	seq2seq	. se	dedding.mn.seg2seg/embedding.	thedding the se
				0 2010	chatbol.cpython-35.pyt	seq2seq	58	bedding mn seg2seg/RNN/Embe	thedding mn se
				CI 117770	C chatbot.cpython-35.pyc	seq2seq.	5.0	whedding me sep2seq/embedding	sbedding enn se
						100 B		and the second	
0 #	intes III Report	D Text D	Her = De		C D Transcript P Picture D P	ieview 🖽 Console	D.	Look of Dr Condition 9 Filter	P BrScillet
					Find Next 1 + + + F Corr			E C / Sct > New	# Colore II
1350- 1357 2414 2414 2414 2425 2425 2425 2425 2425	Contration mode sodel support sensett verd antedding fedinget As I t singlest reserve themsing feding	ddinge wid Uses nie G minestingk s frum 4s 141_setrio 8 sf und_wilton wf complem	b prestraine coglelines pr r-encodeing r est finet erfs 6, fy ch f eff finet fi ch f	e rrind ver reind ver reindere	M. S. Santa and S. Santa and Santa and Santa and Santa a	<ul> <li>maintaing weights of the presidenting real provides and subscriptor - 2,000 drypel real set : 100,000 C presented set: - 2,000 C p</li></ul>	n/mbes n/mbes ng pos- formet formet sf Tharm spint0-m spint0-m	loaded bing wr transon * for we many transfer to the transfe	Į

Figure 11 - seq2seq pyc

The keyword seq2seq was also found in the model.ckpt file of our chatbot. This also confirms that the training of an ANN was committed. Since we know that model file only generates once you start training a neural network.

	5 Extence * P			Ina Search Re	suits - A Bookmark - 5 Go	offe 1 % Tars + D Revent	aciaer -	III fesiks +   # Acake +	
	C Selected 0.4		2 8		O Timeline 🕮 dallery				
	tepression	Items	No	E + B	+ □ Selected 0/277875				
	<ul> <li>seq2seq</li> <li>Textorflow</li> </ul>	208	277,825		Name	Expression	Hit Text	Codepage Preview	
0.3	a nettix	506	1.522	13 95200	C READMELmarkdown	seq2seq	56	oputidhib thutjeq2seq.twitter.tar.g	cpu8dhdstlu/se
0.4	# gts/1070	1	4	CI 95291	D basic_decoder.py	perspect	38	rflow/contrib/seq2seq/gython/ops	effend contribute
				CI 99202	C helper.py	peqSpeq	28	rflow/contrib/seq2seq/python/ops	-rflowicontrib/se
				CI 15303	C losspy	perspec	w	eflow/contrib/seq2seq/python/ops	rflow/constitute
				0 55224	C1_Int_py	seq2seq	56	ontribulegacy_seq2seq/_initpy	onerits/legacy.se
				CI 95225	D_int_py	perform	18	rflow/contrib/seq2seq/_init_py	wflow/contrib/se
				12 15/06	C model.ckpt	seq2seq	58	shedding_mm_seq2seg/embedding_	ibedding_rin_se
				15207	🗋 model.ckpt	seq2seq	58	sbedding_mm_seq2seq/embedding_	thedding_ran_se
				4	1				
11.7	etca III Report	G Text   G	Hex E De	este B Da	ic (i) Transcript (II) Picture (i)	Review 17 Conscie	D	Lock of In Condition IP Filter	* EnSeriet .
0.0	intone "L Cones	are + A 1	ert inde -	R Los A	Finc Neal 10 + + IFG	morested Very 8 2		E C / Edt " New	X Daleta II
5 0004 2 0004 2 0004 5 0004 2 0004 2 0004 2 0004 2 0004	00115 / entertin 00250 set/antertin 00565 ecoder/au1 00175 decr/sm_0to 00175 decr/sm_0to 00175 decr/sm_0to 00175 decr/sm_0to 00175 decr/sm_0to 00125 / estertin/g	<pre>F readedii ng pnn deci i rnn cell/ Lii rnn cell/ cder/maiii r/rnn decode con decode</pre>	Sor/emessi /sell_d/see ll/cell_l/see eth_cell/co eth_c	<pre>local ( solution ) imp/Adam_1 im_1erm_0ell im_1erm_0ell im_1erm_0ell im_mell/dell im_mell/dell im_mell/dell imp/mellin pr </pre>	<pre># Beene fing row end b ing row decode /embedding/Adm /dimess d /embedd /dimess d /embedd /dimess d /embedd /dimess /embedde /dimess /embedde /dimess /embedde /dimess /embedde /dimess /embedde /dimess /embedde /embed//embedde /embedde/embedde /embedde/embedde /embedde/embed</pre>	ing ren_segless/empedding re oding ren_segless/empedding re sheateding ren_segless/emp ( restating ren_segless/emp ( restating ren_segless/emp ( restating ren_segless) ( restating rents) ( restating ren_segless) ( restating rents) ( restat	HALL CHOOSE	an deno deficiano en deco deficiano	

Figure - 12 seq2seq .ckpt

# Netflix keyword hit:

I found Netflix keyword hit in some dataset files (.tsv). We can see that the word has been mentioned in a conversation between two parties in the file content.

	enic Tumne									- 0 X
0.00					R To					0
A. D.4	idence + P)	(synand Hit	1.0							
	O view Keyne	ed Hes - 1	8-1- R	н 5	earch	Results - II Bookmark - E G	to fie   * Tags + D Revie	w Package	+ El Results + # Acquire +	
E + 12 5	Selected 0/4		22 1	10	Table	O Timeline III Gallery				
t a	pression	Itarms.	Nite		+ 3	+ + - Selected 0/1522				=
1 2 9	eq2seq ensorflow	20.214	277,875			Nome	Expression	Hit	Codepage Preview	And
1 100		506	1.922		10	3 9435.ttir	netfix	net	e do you watch netflix on ubuntu?	a do you watch netf
4 2 5		346	4.044		17	C 9435.0v	retfix	Ne-	How, to, watch Netflix, 9(20Wetch, L.	How, to watch Net
10.8	anore				10	Th 76125.ttv	netla	Ne-	.0002 PetertA Netto runs on ubu	.0002 PeterFA Net
					11	31994ztv	netfix	net	a how do I run netflix watch now o.	A how do I run net
					20	C 31994aw	resfie	net	ink they have a cetfix application f.	ink they have a netf
				D.	21	2 9279.tav	netfix	ret	Re. how to get netflix to play on u	Ra, how to get net
					12	3279.tsv	netfix	net_	ta_ how to get netflix to play on u	La_ how to get netf
					23	158200.tov	netfia	net	a way to steam netfix in uburtu, th	s way to steam netf
				1						
Feka	# Report 1	Test	Hes = De		. 5	Doc 9 Transcript @ Picture	B Review El Console		Look of P Condition 9 Fits	P Bidone .
-	A Caller					Bernald a all			T O / ict ' Ner	· Delete
25 den 14		1.410 A 1001	flis applia			an <b>Distant</b> watch now an linear Linear et. 2011-12-08118.00:5				)

Figure 13 - Netflix dataset

found Netflix keyword in a dataset.pkl file. The pickle module (.pkl) implements a fundamental, but powerful algorithm for serializing and de-serializing a Python object structure. Tensorflow uses .pkl files when the program is in testing mode to give quick serialized outputs.

Designation of the local division of the loc	Contractor of the local division of the loca	w 🗙 To:	ds 🤏 Enscriet 🏝 AddiEul	dence 🖪 Fat	hmays.			
Keyword Hit	_							
word Hits +	日・三日	ny Spirch P	Results - 🛛 Bookmark -	E Go to file	<ul> <li>Tags •</li> </ul>	Review Pack	age = III. Result	s + 🖉 Acquire +
	0 1	III Table	O Timeline 🖾 Gallery					
Items	Hits	$\Pi \sim 3$	<ul> <li>□ Selected 0/1522</li> </ul>					
208	277,875		Name	Expression	Hit			Codepage
20,214	2,708,563							Preview
506	1,522		A STATE OF A					
		13,125		netflix	Netfix			
		□ 126	🗋 dataset-ubunbu-10 pid	netflix.	netfix:	-with-ease-us	rg-unetbootin/"A	IctE netflix MictE wind
		1	a second second					
Gal shudow pportable") "BOG store rg/rg2/youtu qwrty"Nicl w.makeusect	rtikiz → fr Ref-uright "MÖSZ plum Re-SL/"HÖst Fuellf"Nact Loss/tag/1	gDieilan Wict ogen "MCCE imi Cochas"HC Liset"Nic Stall-lin grub.enbu	NEWE foolish "NCOE thinks N1""MICE instat"MICE v a V"NOCE //INTL.et/p/n1 2 VE ademos"NTOE alagiat " G101"MacE Join"MpcE 1 2	Call The State of	fright (an Lromann"M E youtupe "IBoE key E mikrud"	Anne-Mid (/is Bet - winder M -0119908 pare sep1608 keye p NécE ata-1.50	eptila.org/2 C.accesspoin heses MOSL/ sThiot A. Mi NACT addreys	Default User
	Incred Hits * Items 206 20.214 506 Page * A T ph/200604/9 Nonct - N Bodg * A A ph/200604/9 Nonct - N Bodg * A A Ph/200604/9 Bodg * A Ph/200604/9 Ph/200604/9 Bodg * A Ph/200604/9 Ph/2004	acid His + E + F + F + F + F + F + F + F + F + F	word Hes         +         Fair Spect           Image: C         Image: The Spect Spe	excret His + B + I and Sensiti ResLas - B Boormak +           B         B         Table Sensiti ResLas - B Boormak +           B         B         B Table (0) There B & Golery           B         B         B Table (0) There B & Golery           B         B         B + B + C         B sectod 3152           200         277375         Name         B sectod 3152           500         1,522         1.14         D memory/fill           130         1 dataser shorton 100 pill         1.35         G dataser shorton 100 pill           131         1 dataser shorton 100 pill         1.35         C compression 100 pill           1325         1 dataser shorton 100 pill         1.35         C compression 100 pill           1335         1 dataser shorton 100 pill         1.35         C compression 100 pill           1335         1 dataser shorton 100 pill         1.35         C compression 100 pill           1335         1 dataser shorton 100 pill         1.35         C compression 100 pill           1335         1 dataser shorton 100 pill         1.35         C compression 100 pill           1335         1 dataser shorton 100 pill         1.35         1.35         1.35           1335         1 dataser shorton 100 pill         1.35         1	eard Has + B +   → Tau-Senth ReLLs - E Boornah + S Go to file	and Hts + B + A Rev Spech ResLts + B Boomerk + S Gots Rel ♥ Trgs + Terms Nite S = Trgs + B Frede   0 Treater = Galaxy 200 277273 20214 2,700,500 1742   1 Arrivel S = Galaxy 20214 2,700,500 1742   1 Arrivel S = C = C = C = C = C = C = C = C = C =	eard Has + B +   A faux Search Routes + B Bookmark + S Go to Re   ● Tags + D Reven Fack Terms Intern B   ■ Table   O Timere ® Golery Terms Intern B   ■ Table   O Timere ® Golery Terms Intern B   ■ Table   O Timere ® Golery 200 277375 500 277475 500 277475 500 277475 500 277475 500 277475 500 27	ecto His + B + A Raw Saunch Reubs + B Boornak + S Gotoffer ♥ Tags + D Review Ractage + B Result Teams Hits 2008 277.873 2024 270575 2024 270575

Figure 14 - Netflix pkl

One interesting find for this keyword was in TrainingRAMdump file that was captured while program was training. This artefact confirms that the dataset that contained this keyword was also used in training the program.

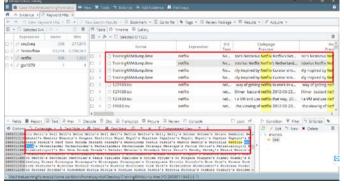


Figure 15 - Netflix RAM

Based on the artefacts I found regarding Netflix keyword, I can confirm that string 'netflix' was a part of dataset and the dataset was used while the program was training.

# Gtx1070 keyword hit:

The keyword GTX1070 was found on SWAP partition. Since the SWAP partition is considered as unused disk area, EnCase shows it as one single raw file.

SWAP file is used for paging. So since the keyword is mentioned here along with strings like tensorflow in the content, we can conclude that GPU acceleration was used to train neural networks using tensorflow.

	Keyword Hi				-	1						
				nchillesuta - 11 Bookmarl		e 🖣 Tap	s = 10 Review	Fackage + III Results	+ - Acquire	-		
+ Seected 0/4				able O Timeline 🕮 Galle	7							
Expression	itema	Hita		<ul> <li>Ib + ID Selected 014</li> </ul>								-
Tensorflow	208	277.875		Name	Expression	Hit Text			depoge extern		_	
11.2 wills	506	1.522	11	III Unused Disk Area	gtx1070	GERIGZO	4960 ° 6	22mineGeforceGI	x1070	1. 16	22maneGe_	
gtx1070				Ill Unused Disk Area	gtx1070	GEK1070	POUL A	22nameGeforceGit	K1070,1 a#D0	Ger 0	dimajore.	
- parters			03	El Unused Disk Avea	gtx1070	GEX1070	4D60 * 6	22 nameGeForceGT	K1070 #8060		22 parteGe	
			0.4	III Uraned Disk Area	gtx1070	GEX1070	8860 A	22.namelieforce01	K1070_1 +FOU	0.440	Consider.	
			*									
			tode	II Dec El Transcript III s #4 Find Next III *				Luci el	P Condition		The Enderset	•

Figure 16 - gtx1070 in SWAP

#### VI. CONCLUSION

Building an ANN with the help of machine learning has got better with the introduction of Google's open source machine learning library TensorFlow.

After finding the relevant artefacts in the investigation of the evidence images, I can conclude that a machine learning program based on tensorflow was trained as well as performed on the system.

These findings can be used as a reference in future cases to detect or identify the use of machine learning libraries, algorithms, techniques etc.

However, a lot of research work still needs to be done in this field. Proper and deeper analysis of volatile information would be beneficial as well as more in-depth analysis of neural networks might help us to get more familiar with machine learning programs in the scope of digital forensic.

#### VII.REFERENCES

- [1] https://venturebeat.com/2017/05/18/ai-weekly-google-shiftsfrom-mobile-first-to-ai-first-world/
- [2] https://techcrunch.com/2017/05/17/the-tensorflow-researchcloud-program-gives-the-latest-cloud-tpus-to-scientists/
- [3] https://techcrunch.com/2016/08/05/carnegie-mellons-mayhemai-takes-home-2-million-from-darpas-cyber-grand-challenge/
- [4] http://www.zdnet.com/article/how-ai-powered-cyberattackswill-make-fighting-hackers-even-harder/
- [5] https://github.com/Conchylicultor/DeepQA
- [6] http://www.cs.cornell.edu/~cristian/Cornell\_Movie-Dialogs\_Corpus.html
- [7] https://en.wikipedia.org/wiki/Word\_embedding
- [8] https://github.com/504ensicsLabs/LiME
- [9] https://rufus.akeo.ie/
- [10] https://apps.ubuntu.com/cat/applications/precise/gnome-diskutility/
- [11] https://www.guidancesoftware.com/encase-forensic