DIDarknet: A Contemporary Approach to Detect and Characterize the Darknet Traffic using Deep Image Learning

ARASH HABIBI LASHKARI, GURDIP KAUR, and ABIR RAHALI, Canadian Institute for Cybersecurity, University of New Brunswick, Canada

Darknet traffic classification is significantly important to categorize real-time applications. Although there are notable efforts to classify darknet traffic which rely heavily on existing datasets and machine learning classifiers, there are extremely few efforts to detect and characterize darknet traffic using deep learning. This work proposes a novel approach, named DeepImage, which uses feature selection to pick the most important features to create a gray image and feed it to a two-dimensional convolutional neural network to detect and characterize darknet traffic. Two encrypted traffic datasets are merged to create a darknet dataset to evaluate the proposed approach which successfully characterizes darknet traffic with 86% accuracy.

CCS Concepts: • **Networks** → *Network security*; • **Security and privacy** → *Intrusion detection systems.*

Additional Key Words and Phrases: darknet, darknet traffic, encrypted traffic, VPN, tor, deep learning, detection, characterization

ACM Reference Format:
Arash Habibi Lashkari, Gurdip Kaur, and Abir Rahali. 2020. DIDarknet: A Contemporary Approach to Detect and Characterize the Darknet Traffic using Deep Image Learning. 1, 1 (December 2020), 21 pages.

1 INTRODUCTION

Darknet is the unused address space of the internet which is not speculated to interact with other computers in the world. It is named dark because of its anonymous nature, virtual marketplace and cryptocurrency [44]. Any communication from the dark space is considered sceptical owing to its passive listening nature which accepts incoming packets but outgoing packets are not supported. Due to the absence of legitimate hosts in the darknet, any traffic is contemplated to be unsought and is characteristically treated as probe [14], backscatter [47], or misconfiguration [21]. Darknets are also known as network telescopes, sinkholes, or blackholes [17].

Different darknets receive significantly different traffic depending on the size of the IP range allocated for monitoring [64]. Even the size of darknet can vary from a single host to the largest available IP address space. To our revelation, the darknet is half the size of what it was postulated to be initially with HTTP as the long-lived service in onions and Zeronet as an ephemeral and emerging component of onions [51]. Nonetheless, darknet offers notable hidden services in its virtual marketplace. Fig. 1 showcases the relationship among various hidden services provided by darknet [65]. For clarity and legibility, the edges are marked with the same color as that of a node for prominent services. Trading tops the list with highest number of closely related services with degree = 23, closeness centrality = 0.661 and page rank (importance) = 0.0725 as listed in

Authors' address: Arash Habibi Lashkari, a.habibi.l@unb.ca; Gurdip Kaur, gurdip.kaur@unb.ca; Abir Rahali, abir.rahali@unb.ca, Canadian Institute for Cybersecurity, University of New Brunswick, Fredericton, New Brunswick, Canada.

XXXX-XXXX/2020/12-ART
https://doi.org/

Table 1. It is followed by technology with 16 related services including trading. The number of triangles represents the intensity of related services in a chain by forming a dependency triangle. For example, trading is related to 23 other services like technology, banks, hacking and security, to name a few. These services form 54 triangles centering at trading.

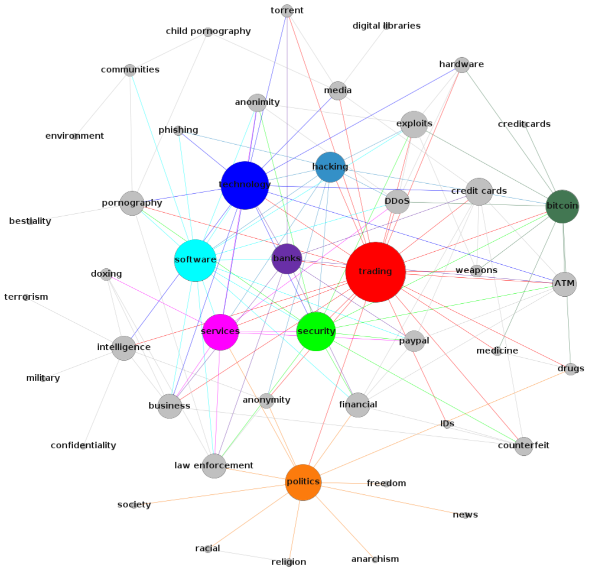


Fig. 1. Hidden services provided by darknet

These hidden services are made anonymous using one of the most approved peer-to-peer anonymizing services called onion routing which provides integrity and confidentiality of data preventing eavesdropping and traffic analysis [28]. On average, almost 2 million worldwide users connected directly and 50K users connected via bridge to Tor services in the first quarter of 2020 [67]. These statistics show the popularity of the Tor network to impart hidden services. Further, merely 20% of these services are suspicious while 48% are benign [5]. A web of darknet hidden services hosted by different websites constitutes the dark web. However, the central focus of research in this paper is darknets, not the dark web.

Analyzing darknet traffic helps in early monitoring of malware before onslaught [29] and detection of malicious activities after outbreak [33]. Following motivations led to this research:

- There are significant efforts on detecting encrypted traffic covering either VPN or Tor traffic separately. None of the papers attempted to combine VPN and Tor traffic in a single dataset that covers a wide range of captured applications and hidden services provided by darknet.
- Detecting darknet traffic in general, and hidden services, in particular, is essential to combat alleged activities before they assault the cyber world.

Following are the primary contributions of this paper:

(1) Most of the existing techniques focused on the classification of darknet traffic while few of them accentuated on anonymized VPN or Tor activity traffic separately. This paper puts forward a novel technique to detect and characterize both VPN and Tor applications together as the real representative of darknet traffic.

Table 1. Relationship Analysis among Hidden Services

Service	Degree	Closeness	Triangle	Page Rank
trading	23	0.661	54	0.0725
technology	16	0.548	37	0.0491
software	15	0.505	35	0.0456
security	14	0.517	36	0.0414
services	12	0.523	27	0.0371
politics	12	0.505	6	0.0563
bitcoin	11	0.473	19	0.0368
hacking	10	0.483	23	0.0307
banks	10	0.478	20	0.0305
credit cards	10	0.468	16	0.0284
exploits	9	0.473	20	0.027
business	8	0.483	13	0.0263
DDoS	8	0.473	17	0.0247
ATM	8	0.459	17	0.0245
financial	8	0.463	13	0.0257
law enforcement	8	0.478	9	0.0273
intelligence	8	0.463	5	0.0359
pornography	8	0.483	9	0.030
paypal	7	0.424	13	0.021
media	6	0.459	5	0.0239
counterfeit	6	0.445	6	0.0204
anonimity	6	0.441	11	0.0192
anonymity	5	0.478	4	0.0192
hardware	5	0.432	7	0.0166
weapons	4	0.432	4	0.0140
drugs	4	0.45	4	0.0162
torrent	4	0.445	5	0.0145
doxing	4	0.394	3	0.0154
communities	4	0.371	2	0.0192
medicine	3	0.412	3	0.0122
child pornography	3	0.348	1	0.0139
phishing	3	0.394	3	0.0110
IDs	2	0.409	1	0.0088
racial	2	0.340	1	0.0126
religion	2	0.340	1	0.0126
society	1	0.338	0	0.0072
freedom	1	0.338	0	0.0072
anarchism	1	0.338	0	0.0072
news	1	0.338	0	0.0072
military	1	0.319	0	0.0070
confidentiality	1	0.319	0	0.0070
terrorism	1	0.319	0	0.0070
bestiality	1	0.328	0	0.0064
environment	1	0.272	0	0.0073
digital libraries	1	0.316	0	0.006

(2) Amalgamating two public datasets to create a complete darknet dataset covering VPN and Tor traffic.
(3) Demonstrating the effectiveness of two-dimensional convolutional neural network to detect and characterize darknet traffic with high detection rate that encourages identifying diverse and suspicious hidden services.

Rest of the paper is organized as follows: Section 2 details the related works on darknet traffic detection along with limitations. Available traffic datasets are analyzed in Section 3 which is followed by our combined dataset details in Section 4. Background of convolutional neural networks is presented in Section 5. The proposed model to carry out research in this paper is elaborated in Section 6. Experiments are detailed in Section 7. Section 8 presents a deep analysis of darknet traffic detection and characterization. Finally, Section 9 concludes the paper with future directions.

2 RELATED WORKS

This section introduces the recent work in darknet detection while shedding light on encrypted, VPN, Tor and darknet traffic detection in chronological order. We also list limitations of existing work and proposed solution to tackle most of them towards the end of this section.

2.1 Darknet Traffic Detection

Classifying network traffic in the dark dates back to early 2000s with initial works using decision trees [31], single-flow traffic [72], size and direction information for the first four to five data packets of a connection [12, 13], session flows [68], taxonomy and traffic rules for various activities in darknet [39], and correlating low-interaction honeypot with darknet traffic [3].

In 2015, Nishikaze et al. [50] captured 303,733,99 packets in communication between subnets at source network and dark network. They created twenty-seven categories of traffic analysis profile in a 27-dimensional feature vector consisting of packet count, source IP and port, destination IP and port. Malicious packets were identified by performing hierarchical clustering and matching malware signatures with identified packets. However, they were unable to identify new malware samples due to capturing at the local tap. Distributed Reflection Denial of Service (DRDoS) attack was detected by extracting additional information such as intensity, rate and geo-location [20]. In another attempt, the total frequency of packet, number of source hosts, and targeted port for TCP and UDP protocols were used to detect darknet traffic [25].

In 2016, Ban et al. [11] grouped similar attack patterns collected from the darknet and used a time-series to characterize the activity level of these attack patterns. A comprehensive survey also reported the use of Honeyd (low interaction honeypot) to deploy darknet services and time-series techniques to analyze darknet traffic [22].

In 2018, Wang et al. [69] improved the existing user profiling methods by extracting sensitive personal information such as top names, number of attributes, email domain and geographic distribution from the darknet. Eight hidden marketplaces on the darknet were explored to create a threat dictionary for mining text to identify new threats [19].

In 2019, Pour et al. [53] proposed a novice approach to apprehend attackers' behavior, the width of the darknet viewpoint, probability of detection and minimum detection time by using stochastic modelling. However, the work was unable to infer large-scale distributed probes within a stipulated time, similar to previous works. Similar to [50], Montieri et al. [46] classified Tor, I2P and JonDonym traffic using a hierarchical approach achieving 75.56% f-measure.

2.2 Encrypted Traffic Detection

Researchers have reported some representative works in classifying encrypted traffic [4, 6, 7, 9, 35, 66, 75].

In 2011, Alshammari et al. [55] showed that encrypted traffic can be identified without inspecting payload, port numbers and IP addresses. Authors used flow-based and packet header features to classify SSH and Skype traffic with AdaBoost, GP and C4.5 to infer that flow-based features performed better than packet header features. Gu et al. [73] proposed flow-based online internet encrypted traffic (Skype) identification method by using machine learning obtaining high accuracy and low overhead. Kolton et al. [34] provided an encrypted intrusion detection system capable of decrypting SSL and other encrypted traffic. On the contrary, Sherry et al. [63] developed a blind box system called deep packet to detect encrypted traffic without decrypting it.

In 2017, Shen et al. [62] captured SSL/TLS flows from 14 real-time applications including Airbnb, Alipay, Baidu, Blued, Ele, Evernote, Facebook, Github, Instagram, LinkedIn, NeteaseMusic, Twitter, Weibo, and Yirendai. Certificate and application data is extracted to form a bigram to classify encrypted traffic with Second-Order Markov chain fingerprint considering certificate packet length and application attribute Bigram. Bigram showed better results with an improvement of 29% true positive rate and fall by about 25% in false-positive rate. Wang et al. [70] used one-dimensional and two-dimensional convolutional neural network (CNN) to conclude that 1D CNN is better than previous best C4.5 and 2D CNN for encrypted traffic in ISCXVPN2016 dataset. The paper achieved a precision of 85.38% and 92% for non-VPN and VPN traffic respectively.

In 2019, unlike [34, 63], Shekhawat et al. [61] emphasized feature selection using recursive feature elimination and tested their results with support vector machine, random forest, and XGBoost. Authors insisted on obtaining minimal feature set instead of an optimal feature set in their major findings. Lotfollahi et al. [41] in 2020 presented "Deep Packet", a framework to automatically extract features from network traffic using 1D CNN and Stacked AutoEncoder (SAE). It outperformed earlier state-of-the-art results to come up with 98% accuracy in identifying applications and 93% accuracy in characterizing traffic beating all machine learning, 2D CNN and SAE algorithms.

2.3 VPN Traffic Detection

In 2010, Nadeau et al. [49] proposed a traffic flow labelling process for edge routers to export label binding information to conquer shortcomings related to losing granularity between multiple routers in traditional MPLS technology.

In 2014, Mullick et al. [48] intercepted VPN traffic by identifying application requesting access to a device. The device associates an authorization policy with the application process and a decision to allow or deny access to the resource is taken based on application identification and authorization policy. An agent on client-side determines the destination (based on port and network identifier) specified by application under question and signals communication interception from another network. Patel et al. [52] used first and second digital certificates to identify and classify VPN traffic respectively. A secure key exchange is initiated by sending an encrypted first digital certificate and plain second digital certificate to the second network device. The second digital certificate is used and stored by an intermediate network device to interpret classification information associated with the first network device.

In 2016, Draper-Gil et al. [27] generated a representative dataset, ISCXVPN2016, to capture encrypted and VPN traffic and characterized it using time-based features. They used C4.5 and kNN to train multi-class classifiers to characterize VPN traffic into seven different categories with 80% accuracy. Bagui et al. [10] used the same dataset and classified it with six machine learning models to identify best-supervised model to distinguish VPN and non-VPN traffic. It was found that RF and GBT outperformed all other supervised models.

In 2018, Miller et al. [42] used multi-layer perceptron to detect encrypted VPN traffic captured by using Wireshark and NetMate. The authors used TCP flow-based features to classify incoming traffic to a web server into OpenVPN or non-VPN with an accuracy of 92% and 93% respectively. Caicedo-Muñoz et al. [15] extended the work done by [27] by adding a quality of service (QoS) classifier and per-hop behavior (PHB) marking scheme for labelling traffic in a specific application domain. They divided ISCXVPN2016 dataset into two datasets containing separate VPN and non VPN traffic in the first dataset and combined traffic with PHB labels in the second dataset. Flow timeout values (15s, 30s, 60s, 120s) were used for testing their technique. They compared their results with [27] to come up with 94.42% and 92.82% of accuracy for classifying Non-VPN and VPN traffic respectively.

2.4 Tor Traffic Detection

In 2011, Wang et al. [71] exploited HTTP's vulnerability to launch HTTP-based application-level man-in-the-middle attack against Tor and proposed effective countermeasures to thwart it. In another attempt to identify anonymous services provided by Tor, several protocol-level attacks targeting entry onion router to duplicate, modify, insert, or delete cells of a TCP stream from a sender to cause cell recognition errors at the exit onion router were launched [38]. Tor introduces a delay in data transmission owing to several onion routers used to increase anonymity [40]. However, LASTor (Tor client) was developed to achieve significant latency gains and protect against snooping without changing Tor relays [2]. Identifying the type of application based on flow-based burst traffic facilitates revealing encrypted data flowing through the onion routers [30, 59].

In 2015, TorWard [37] was proposed to detect and classify malicious traffic over Tor by passing it through an IDS at Tor exit router. Authors recorded a total of 3,624,700 alerts including P2P, botnet, spam, and other malware traffic. using Tor for anonymizing hidden services in the darknet [43]. In 2018, Montieri et al. [45] classified traffic of anonymity tools Tor, I2P, and JonDonym using Anon17 dataset and revealed that these anonymity tools are truly distinguishable. Saleh et al. [56] carried out a comprehensive survey on classifying Tor traffic, quantification, and comparison of various techniques used for deanonymization, selecting the path and improving the performance of encrypted traffic in the darknet.

Several eavesdropping attacks are launched on anonymous Tor network ranging from measuring the battery usage of a smartphone to track the websites visited [74] to adaptive stream mining for website fingerprinting [8] and grabbing foreground image information to classify images [24]. Tor browser eventually leaves multiple artifacts and traces on user machine in system memory [1]. To deepen the effect of information leakage, bitcoin payment made by users of Tor hidden services is also disclosed to breach user privacy [32].

2.5 Limitations of Current Detection Techniques

Table 2 presents pros and cons of leading darknet traffic detection techniques. Current techniques cover (1) darknet traffic which primarily focuses on classification, (2) encrypted traffic which is based on classification and feature extraction, (3) VPN traffic which contains classification and authorization, and (4) Tor traffic which emphasizes anonymizing Tor services and launching attacks to breach Tor network. Following shortcomings are found in existing literature work:

Table 2. Pros/Cons of Leading Darknet Traffic Detection Techniques

Paper	Technique	Pros/Cons
Alshammari et al., 2011 [55]	Signature-based	Encrypted traffic can be identified without inspecting payload and packet header, Technique can be evaded by randomly padding packet payload throughout the connection
Gu et al., 2011 [73]	Flow-based	High accuracy and low overhead
Nishikaze et al., 2015 [50]	Hierarchical Clustering	Cannot detect new malware, Threshold for hierarchical clustering is undefined
Sherry et al., 2015 [63]	Deep packet inspection directly on the encrypted traffic	Performs best over long-running, persistent connections
Ban et al., 2016 [11]	Time-series	Early detection of novel attack patterns
Draper-Gil et al., 2016 [27]	Time-series	Generated VPN traffic dataset and evaluated it with 80% accuracy in VPN traffic characterization
Shen et al., 2017 [62]	Second order Markov Chain	Detected real-time applications with 29% improvement in earlier models
Wang et al., 2017 [70]	CNN	High precision for classifying VPN and non-VPN traffic
Wang et al., 2018 [69]	User Profiling	Accurate extraction of top user names in darknet
Miller et al., 2018 [42]	Multi-layer Perceptron	On an average 92% accuracy
Caicedo-Muñoz et al., 2018 [15]	QoS classifier and PHB marking scheme	Improved the accuracy achieved by [27] to 92.82% for classifying VPN traffic
Saleh et al., 2018 [56]	-	Comprehensive survey on Tor traffic classification
Pour et al., 2019 [53]	Stochastic Modeling	Minimum detection time
Montieri et al., 2019 [46]	Hierarchical Approach	Classified anonymized traffic with 75.56% f-measure
Deep Packet [41]	CNN and SAE	98% accuracy in application identification and 93% accuracy in VPN traffic characterization
DeepImage	CNN	Classification and characterization of diverse hidden services and applications supported by Tor and VPN, analysis of IP-based darknet traffic, good accuracy

- Darknet traffic detection is centered around gathering parameters from flows and connections to identify malicious samples. Minimal work on feature extraction, user profiling and attack pattern characterization is also reported. Nevertheless, deanonymizing hidden services provided by darknet marketplace remains an untouched area.
- A couple of research papers used flow-based and packet header features to classify encrypted applications such as Skype, SSH, and real-time applications including social networking, search engines, note-making, and media. However, diverse applications and protocols like browsing, chat, email, file transfer, streaming, VOIP, and torrent are not incorporated in the majority of past papers.
- Primary focus is laid on classifying VPN traffic using flow-labelling and digital certificates. None of the papers characterized VPN and Tor traffic.
- Deanonymizing Tor services and applications by using application-level and protocol-level attacks, identifying hidden servers, classification, and eavesdropping are the fundamental research areas on Tor. Less attention is paid to deanonymize IP addresses involved in providing hidden services.

To prevail over diversity, hidden services and combines Tor and VPN traffic, we propose a two-dimensional convolutional neural network that uses feature selection to shortlist the finest set of header and payload features from combined ISCXVPN2016 [27] and ISCXTor2017 [36] dataset. The feature vector is transformed into gray images to feed to the proposed model to detect hidden services and characterize a diverse range of benign and anonymized (VPN and Tor) applications in darknet.

3 ANALYSING AVAILABLE TRAFFIC DATASETS

This section analyzes available datasets in chronicle order and proposes an evaluation criterion to assess them, and finally highlights their major issues and shortcomings.

3.1 Publicly Available Related Datasets

One of the most significant requirements of training a model is the availability of a representative and comprehensive public dataset. Although a limited number of public datasets are available, none of them is unanimously agreed upon by researchers to classify encrypted traffic [54]. We present a taxonomy for listing and explaining the publicly available datasets in this sub-section.

DARPA - MIT Lincoln Laboratory (1998-99): This is one of the conventional datasets that covers twenty-seven attack categories and normal background data captured over seven weeks. The dataset captured FTP, telnet, SNMP and browsing activities along with buffer overflow, Nmap, syn flood and denial of service attacks, to name a few. It includes network traffic and audit logs collected on a simulated network thereby lacks real-time attack traffic. Dataset is evaluated both in online (Air Force Research Lab) and offline mode (simulated network).

CTU-13 - Czech Technical University (2011): It was captured to collect real-time botnet traffic along with background and normal traffic. This dataset includes thirteen malware traffic scenarios corresponding to different botnet samples. The malware traffic consists of bidirectional flows captured by executing a particular malware in a Windows virtual machine and recording the network traffic generated on the host. Netflow files store distribution of various botnet flows, C&C flows, background flows and normal flows.

Malware Capture Facility Project - Czech Technical University (2013): The objective to generate this dataset was to capture, analyze, and publish real malware network traffic over several months for some cases. Windows virtual machines are hosted on Linux machines to execute malware to prevent DDoS and spam emails. Traffic is labelled to facilitate ease of use.

Anon17 - NIMS Lab (2014-17): This dataset consists of three anonymity tools: Tor, I2P and JonDonym. Captured in a real network environment, the dataset is labelled based on information available on chosen anonymity services. It contains Tor, TorApp, TorPT, I2PApp80BW, I2PApp0BW, I2PUsers, I2PApp, and JonDonym data.

ISCXVPN2016 - ISCX (2016): It captured real-time VPN traffic for applications such as web browsing, chat, file transfer, email, streaming, VOIP, and P2P using Wireshark and TCPdump. An external VPN service provider was used to generate encrypted traffic which was labelled to facilitate ease of use.

ISCXTor2017 - ISCX (2017): It captured real-time Tor traffic for applications such as browsing, chat, FTP, email, audio and video streaming, VOIP, and P2P using Wireshark and TCPdump. The dataset is labelled to facilitate ease of use.

Darknet Usage Text Address (DUTA)-10K - GVIS Lab (2019): It comprises twenty-five categories of legal and illegal activities with over 10,367 manually labelled onion domains. To follow up with the most recent hidden activities on Tor, it introduced CryptoLocker, a new category which has spread widely after WannaCry ransomware.

3.2 Dataset Evaluation Criteria

We define six-fold evaluation criteria based on [16, 23, 60] to compare publicly available encrypted or darknet traffic datasets used in the past as:

(1) Covering Different Connections (CDC): The first and foremost criteria is whether the dataset contains Tor (T), VPN (V), or Tor over VPN (TV) traffic.
(2) Complete Traffic (CT): It includes a diversity of protocols (DP) and diversity of applications (DA) used to capture complete network traffic.
(3) Complete Interaction (CI): It covers complete interaction with different protocols for sending and receiving a variety of data such as Audio (A), Video (V), File transfer (FT), Text/Chat (T), Email (E), VoIP (Vo), Web Browsing (B), and P2P (P2).
(4) Complete Capture (CC): By capturing header (H) and encrypted payload (P) without anonymization (A), it is ensured that dataset remains transparent to researchers by revealing every bit of captured information.
(5) Feature set (FS): It represents features stored in the dataset. We classify this criterion as header and payload features to synchronize with complete capture mentioned above.
(6) Metadata (M): It means that details about dataset like captured traffic, attack scenario, type of protocols etc. are made available.

Table 3. Dataset Evaluation

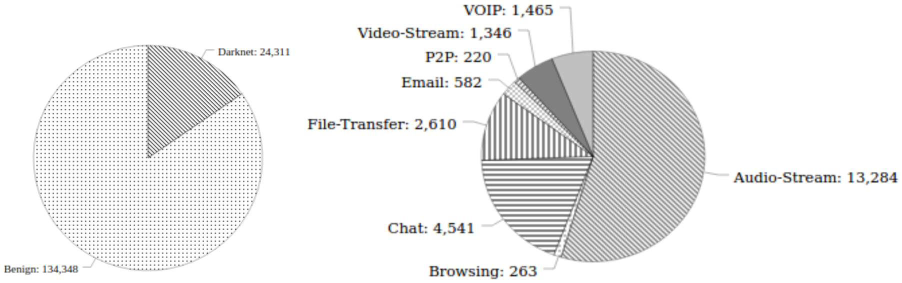
Dataset	CDC			CT		CI								CC			FS		M
	Tor	VPN	TV	DP	DA	A	V	FT	T	E	Vo	B	P2	H	P	A	H	P	
DARPA	N	N	N	Y	N	N	N	N	N	Y	N	Y	N	Y	N	N	Y	N	Y
CTU-13	N	N	N	N	N	N	N	N	Y	Y	N	N	Y	Y	N	N	Y	N	Y
MCFP	N	N	N	N	N	N	N	N	Y	Y	N	N	Y	Y	N	N	Y	N	Y
Anon17	Y	N	N	Y	Y	N	N	N	Y	N	N	N	N	Y	Y	N	Y	Y	Y
ISCXVPN2016	N	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
ISCXTor2017	Y	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y	Y
DUTA-10K	Y	N	N	Y	Y	N	N	N	N	N	N	Y	N	N	-	-	-	-	N

Based on these criteria, we present a detailed comparison of seven related datasets in Table 3. Most of the datasets covered diversity in protocols and applications and prefer not to anonymize captured traffic. Despite containing encrypted or anonymized traffic, none of these datasets is complete in one way or the other to detect and characterize darknet traffic except ISCXVPN2016 and ISCXTor2017 that fulfil most of the defined criteria.

4 DATASET

Based on outcomes of dataset evaluation criteria, we chose ISCXVPN2016 [27] and ISCXTor2017 [36]. Both datasets captured regular, VPN and Tor traffic for seven diverse categories under respective applications: Browsing (Firefox and Chrome), Chat (ICQ, AIM, Skype, Facebook and Hangouts), Email (SMTPS, POP3S and IMAPS), File Transfer (Skype, FTP over SSH (SFTP) and FTP over SSL (FTPS) using Filezilla and an external service), Streaming (Vimeo and Youtube), VoIP (Facebook, Skype and Hangouts voice calls), and P2P (uTorrent and Transmission (BitTorrent)).

We combined these datasets to create a new two-layered traffic dataset and named it Darknet dataset to further refer in this paper. The first layer of Darknet dataset is labelled as benign to present regular traffic. The second layer is labelled as darknet to represent anonymized (Tor or VPN) traffic related to hidden services provided by darknet. We segregated streaming traffic into audio and video for second layer labelling to make eight categories. Fig. 2 shows different protocol traffic along with the number of records in the Darknet dataset. The Darknet dataset consists of 158,659 records in total. There are 134,348 benign samples and 24,311 darknet samples. Audio-Streaming has the highest number of 13,284 samples whereas minimum samples are captured for P2P protocol in this dataset.



(a) Layer1

(b) Layer2

Fig. 2. Number of hidden services-based activity

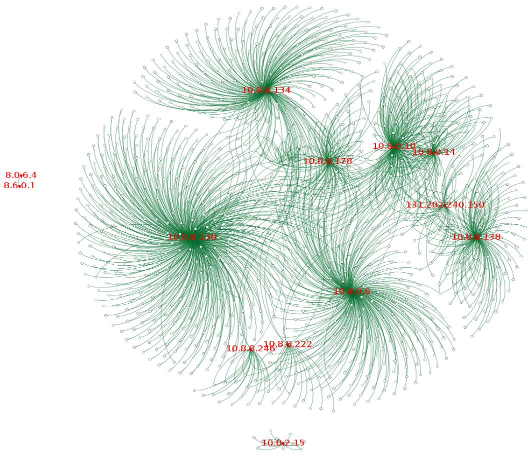


Fig. 3. Communication between unique pairs of source and destination IP addresses

To analyze the activities captured in the dataset, we plotted a sample communication graph between unique pairs of source and destination IP addresses using Gephi [26] as seen in Fig. 3. The intensity of this directed graph admits the extreme communication that took place between distinct hosts while generating the dataset. The graph highlights only the top 10 source machines sending data to the maximum number of destination machines. Allegedly, most of the machines have private IP addresses while only one public IP address host (131.202.240.150) is involved. By examining deeper, it is seen that some of the private hosts used 131.202.240.150 as one of the intermediate machines to transfer data to another private host.

5 CONVOLUTIONAL NEURAL NETWORKS

Convolutional neural network (CNN) is a deep learning algorithm which takes a gray or colored image as input, assigns weights and biases to objects in the image, performs multiple convolutions to reduce the size of the image while retaining important features, and classifies the input image. This section sheds light on layers and functions used in CNN.

5.1 CNN Layers

Three types of layers form the basis of CNN: convolution, pooling, and dense.

1. Convolution: It is a linear operation that uses input image and chosen weight matrix called filter or kernel to produce a convolved image. The kernel repeatedly strides over the input image to compute the sum of the product of selected elements and places the result in the convolved matrix. Multiple convolutions are applied to extract low-level features after the first convolutional layer and high-level features after last convolutional layer. Mathematically, convolution is defined as:

$$f[n, n, n_c]_s * g[f, f, n_c] = h[c, c, n_c] \tag{1}$$

where f is an nxn input image, g is a chosen fxf filter with strides s to produce a cxc convolved matrix h. Input image, filter matrix and the convolved image has n_c channels which are three for RGB and one for a gray-scale image.

2. Pooling: Similar to convolutional layer, pooling layer reduces the dimensionality of the feature vector to lower down the computational power required to process the data. There are two types of pooling: max pooling and average pooling. Max pooling takes the maximum element from the selected portion of the input image with pre-defined filter size and stride value while average pooling goes with the mean value of selected elements to form a pool. Filter size and stride value are called hyperparameters for pooling. Pooling also aids in controlling overfitting while training the model. Mathematically, pooling is represented as:

$$n = ((n - F)/S) + 1 \tag{2}$$

where n is the size of the image, F is filter size and S represents stride value.

3. Dense: It is also called a fully connected layer where every node in the current dense layer is connected to every other node in the previous layer. Dense layer presents a non-linear operation that transforms input image to multi-level perceptron. The goal of the dense layer is to tune the weight parameters to create a stochastic likelihood representation of each class.

5.2 Functions

Following functions are used in CNN:

1. Activation: It decides if the neuron would fire or not. Different types of activation functions are available but Rectified Linear Unit (ReLU) produces the best results.

2. Dropout: It is a regularization technique which works by dropping a specific percent of data while training the model to reduce overfitting problem.

3. Flatten: As the name implies, the feature map is flattened into a column to insert this data into an artificial neural network later on.

4. One Hot Encoder: It is a process by which categorical variables are converted into a form that could be provided to learning algorithms to do a better job in prediction.

6 PROPOSED MODEL

This section uncovers DeepImage and explains its two main components: (1) feature extraction to select the best features and (2) layered view of the model.

6.1 Feature Extraction

Feature extraction plays a pivotal role in selecting the best feature set to detect and characterize darknet traffic. It is a two-fold process: (1) pre-processing labelled darknet dataset [27, 36] to extract features and identify target labels; and (2) ranking features using extra trees classifier [57] to find important values for all nodes in the forest of trees and sorting them in descending order to select the feature set with highest importance values. In data preprocessing phase, CICFlowMeter [18] is used to extract 80 network traffic features from the dataset and target labels are assigned to all activity traffic recorded in the dataset to create a feature vector. Out of 80 extracted features, 61 features are shortlisted by using feature ranking as mentioned in feature selection Algorithm 1.

Algorithm 1 Feature Selection

```
procedure COMPUTE_FEATURE_IMPORTANCE($Feature\_Vector$, $Target\_Labels$)
    Build a forest of trees with input (Feature_Vector,Target_Labels)
    for each node in forest do
        Compute $standarddeviation$ of node as array elements
        Sort the node in descending order to get indices and importance values for most important features
        if importance > 0.001 then
            Rank indices and importance values of best features
        end if
    end for
end procedure
```

In the next step, the selected features are used to create a two-dimensional image vector and numerically encode target labels before passing to the ensemble model. The proposed convolutional neural network model constitutes several layers to classify dataset traffic as benign or anonymized in the first layer and to characterize darknet traffic into eight categories as shown in Fig. 4.

6.2 Layered View of Model

Two-dimensional gray image created as a result of feature extraction is taken as input to form an (8,8,1) image vector where 8x8 vector is used to store 61 shortlisted features and third parameter ’1’ presents the gray scale image in the proposed model. It is shown as a flashed view in the ensemble model box in Fig. 4. The input layer of the proposed convolutional neural network is sequential which is followed by first two-dimensional (2D) convolution layer with shape (7,7,32). It applies 32 3x3 filters to extract 3x3 pixel sub-regions with activation function ReLU. The second convolution with 64 3x3 filters (6,6,64) is applied to the model. This is followed by flattening and two dense layers. Flatten layer reshapes the tensor to eight neurons representing eight darknet traffic categories in the dataset. The dense layer is a fully connected layer that connects every neuron in the present layer to every other neuron in the previous layer. Second dense layers contain eight neurons where every neuron points to one class of darknet traffic.

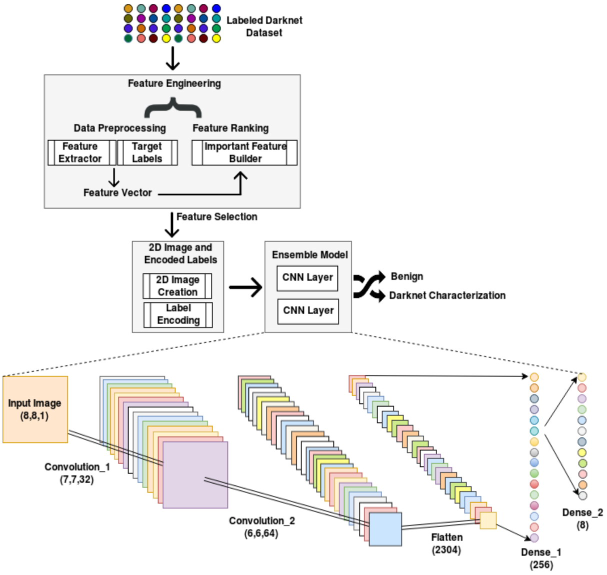


Fig. 4. Methodology architecture

Pooling layers are eliminated from the layered view of the proposed model. It is interesting to mention that adding the first pooling layer after the first convolution layer reduces feature vector to (3,3,64). Therefore, there is less scope of adding more convolutions results in a drastic drop in training and testing the accuracy of the model. Additionally, only 80 features are extracted at the beginning which is shortlisted to 61 after feature ranking and we need only 8x8 input size to accommodate those features.

7 EXPERIMENTS

The proposed system has been implemented in Python using Keras and TensorFlow by utilizing Scikit-Learn [58]. Table 4 shows the parameters with tuned values to obtain maximum accuracy and minimum log loss.

Table 4. Parameters

Parameter	Value
Activation Function (Hidden layers)	RELU
Activation Function (Output layer)	Softmax
Loss Function	sparse_categorical_crossentropy
Optimizer	adam
Epoch	1500
Batch Size	32
Estimators	250
Maximum Depth	16
Early Stopping Monitor	patience = 3

Effect of hyper-parameter tuning to finalize their values for executing the final version of DeepImage is discussed concretely in sub-section 8.6. Experiments are performed on the Ubuntu server with 50 CPUs and 500GB of RAM. Feature extraction is performed on cleaned dataset to shortlist the best feature set from the extracted list. Finally, the dataset is split into a training set (80%) and testing set (20%) to feed to the model.

8 ANALYSES AND DISCUSSION

This section presents the major findings of this research. First and foremost, we bring to light the best feature set to detect darknet traffic and continue with darknet traffic characterization to find the common pattern of activities in darknet traffic in the following sub-sections.

8.1 Best Feature Set

To determine the importance of network features extracted through CICFlowMeter, we eliminated flow label features including flow ID, timestamp, source and destination IP and then computed importance values for all the features (Algorithm 1) as part of the feature extraction process. It is obvious from the list of importance percent of shortlisted features in Table 5 that maximum idle value is the most important feature to detect darknet traffic at layer1. It is closely followed by minimum forward segment size and the minimum backward packet length. At layer2, forward packets per second are the most important feature in characterizing darknet traffic. It is followed by backward packets per second and maximum idle value. Comparison of feature importance values at both layers reveals the following similarities:

- All the shortlisted features contribute almost equally towards training the darknet traffic detector.
- Out of 22 shortlisted features, 15 features are found at both layers which indicates that these features are highly imperative to detect darknet traffic from benign traffic at layer1 and characterize anonymized darknet traffic at layer2.

Table 5. Best Features Selected from Complete List of Extracted Features

(a) Layer1

Rank	Index	Feature Name	Percent
1	F74	Idle Max	0.078017
2	F67	Fwd Seg Size Min	0.075886
3	F12	Bwd Pkt Len Min	0.072589
4	F1	Protocol	0.051608
5	F72	Idle Mean	0.048613
6	F64	Fwd Init Win Bytes	0.042459
7	F42	FIN Flag Count	0.042023
8	F63	Subflow Bwd Bytes	0.039559
9	F40	Packet Length Std	0.036495
10	F11	Bwd Pkt Len Max	0.035476
11	F13	Bwd Pkt Len Mean	0.035112
12	F75	Idle Min	0.034859
13	F53	Bwd Seg Size Avg	0.033452
14	F60	Subflow Fwd Packets	0.032802
15	F65	Bwd Init Win Bytes	0.030974
16	F51	Average Packet Size	0.029259
17	F38	Packet Length Max	0.020524
18	F0	Destination Port	0.018926
19	F39	Packet Length Mean	0.018509
20	F6	Total Len of Bwd Pkt	0.016423
21	F33	Fwd Header Length	0.014669
22	F36	Bwd Packets/s	0.014571

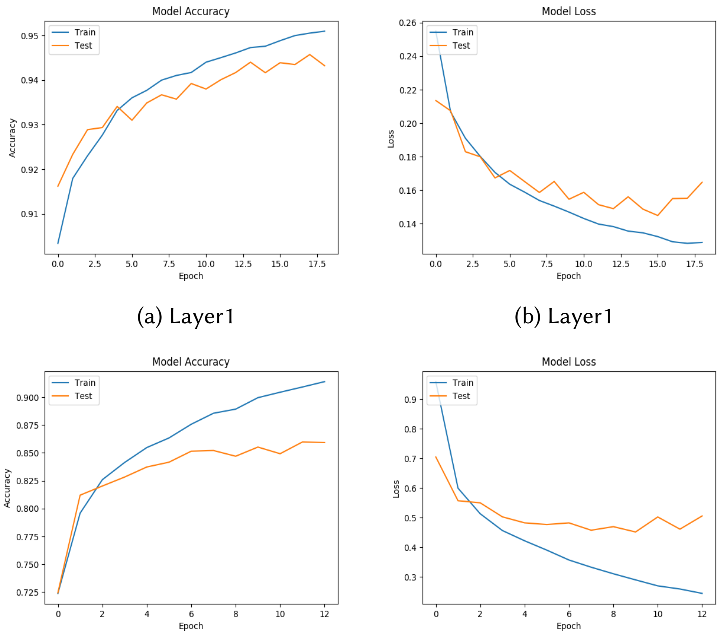
(b) Layer2

Rank	Index	Feature Name	Percent
1	F35	Fwd Packets/s	0.075397
2	F36	Bwd Packets/s	0.062840
3	F74	Idle Max	0.04995
4	F2	Flow Duration	0.04119
5	F15	Flow IAT Mean	0.03928
6	F18	Flow IAT Min	0.03891
7	F17	Flow IAT Max	0.03522
8	F72	Idle Mean	0.03198
9	F75	Idle Min	0.02882
10	F12	Bwd Pkt Len Min	0.02755
11	F63	Subflow Bwd Bytes	0.026458
12	F13	Bwd Pkt Len Mean	0.026138
13	F0	Destination Port	0.025542
14	F40	Packet Length Std	0.024493
15	F53	Bwd Seg Size Avg	0.023945
16	F39	Packet Length Mean	0.023916
17	F51	Average Packet Size	0.023769
18	F11	Bwd Pkt Length Max	0.020243
19	F38	Packet Length Max	0.020210
20	F41	Pkt Len Variance	0.018736
21	F60	Subflow Fwd Packets	0.016200
22	F52	Fwd Seg Size Avg	0.015822

8.2 Accuracy and Log Loss of DeepImage

Best features selected in the previous step are used to create a two-dimensional gray image which is fed to DeepImage for execution. To monitor the performance of DeepImage, we plotted accuracy and logarithmic loss of training and testing curves at different epoch values for layer1 and layer2 respectively as shown in Fig. 5. Following are the main observations derived from accuracy and loss curves which depict the potential of DeepImage to detect and characterize darknet traffic:

- Accuracy curves at both layers show an upward trend which stipulates that with an increase in epoch values, the accuracy of training and testing set is also increasing. There is no sign of overfitting in both the curves.
- Loss curves at both layers demonstrate a downward trend which indicates that with increasing epoch values, log loss of training and testing set is decreasing which is desirable. It is interpreted from loss curves that predicted probability does not deviate from the actual labels in the dataset.
- At layer1, model accuracy for training the classifier is 95% and testing it is 94%. Log loss for training and testing curves goes down to 0.13 and 0.17 respectively.
- At layer2, model accuracy for training the classifier is 92% and testing it is 86%. Log loss for both training and testing curves goes down to 0.2 and 0.5 respectively.



(a) Layer1

(b) Layer1

(c) Layer2

(d) Layer2

Fig. 5. DeepImage: accuracy and loss

8.3 Competitive DL Algorithm

Deep Packet [41], a competitive deep learning approach that used ISCXVPN2016 dataset to classify the VPN traffic with 1D CNN and SAE, outperformed all machine learning classifiers, 2D CNN and SAE algorithms. It obtained 93% accuracy in VPN traffic and 35% in Tor traffic characterization.

Table 6. Comparison with other DL Classifiers

Category	Precision	Recall	F1-Score	Accuracy
1D CNN	0.74	0.73	0.73	0.73
DeepImage	**0.86**	**0.86**	**0.86**	**0.86**

However, we also evaluated 1D CNN using our new darknet dataset which includes both VPN and Tor encrypted traffic. It is found that the accuracy of 1D CNN turned out to be 63% initially and was improved to 73% after applying the feature extraction process and tuning hyper-parameters. We present the results for 1D CNN in Table 6 to compare with DeepImage.

8.4 Darknet Traffic Characterization

We performed multi-class classification and evaluated DeepImage by computing precision, recall, f1-score and accuracy for layer2. It is noticeable from Table 7 that out of 2635 audio-streaming samples passed to detector, 2423 (92%) are identified correctly. Similarly, 38 out of 40 P2P samples are detected as P2P making its recall value 0.98. On the contrary, accuracy for browsing is 47% which is the lowest among all categories. Seemingly, DeepImage fruitfully characterized all darknet samples passed to it with an overall accuracy of 86% with high precision for most of the categories. To prove the performance effectiveness of DeepImage, we compared our characterization results with 1D CNN as mentioned in previous sub-section in Table 6. Obviously, DeepImage performed much better than 1D CNN making it clear that performance of 2D CNN on darknet dataset can not be correlated to encrypted traffic characterization results obtained by previous research such as Deep Packet [41] which uses only encrypted traffic in the form of VPN and non-VPN or Tor and non-Tor traffic.

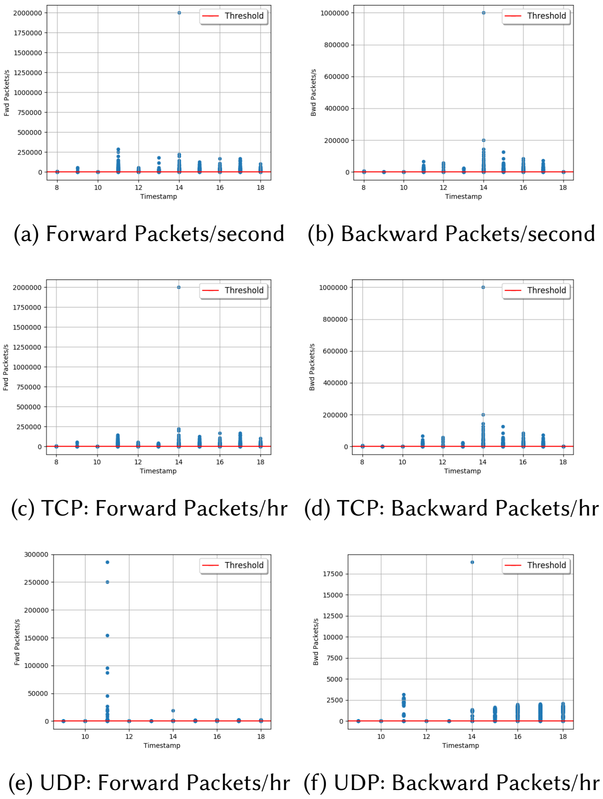
Table 7. Characterization

Category	Precision	Recall	F1-Score	Accuracy	FN Rate	#Testing Instances	#Training Instances
Audio-Streaming	0.92	0.92	0.92	0.92	0.8	2635	10649
Browsing	0.55	0.47	0.51	0.47	0.53	59	204
Chat	0.90	0.86	0.88	0.86	0.14	919	3622
Email	0.66	0.67	0.67	0.67	0.33	124	458
File-Transfer	0.74	0.75	0.75	0.75	0.25	521	2089
P2P	0.90	0.95	0.93	0.95	0.05	40	180
Video-Streaming	0.82	0.88	0.85	0.88	0.12	283	1063
VOIP	0.58	0.61	0.59	0.61	0.39	282	1183

8.5 Analysis of Darknet Traffic

After successfully detecting and characterizing darknet traffic in previous sub-sections, it is pertinent to deeply analyze darknet traffic to find a trend in protocol-wise communication. As listed in Table 5, forward packets/second and backward packets/second is among the top three important features to characterize darknet traffic in darknet dataset. Therefore, we plotted overall hourly traffic and TCP/UDP traffic with respect to time in Fig. 6. Analyzing the trend of forwarding and backward packets/second reveals that most of the time, under 250,000 forward packets and below 200,000 backward packets are received by darknet. Maximum number of forward packets/second reaches 2,000,000 whereas highest count of backward packets/second is 1,000,000. Further, TCP-based forward and backward packets per second bears exactly the same trend as that of overall hourly forward and backward packets. Nonetheless, UDP-based forward packets/second follows a unique trend where the different size of packets is sent at 11:00 AM AST. Overall, the dataset dominantly contains TCP traffic than UDP traffic.

Additionally, we performed source and destination IP address-based TCP and UDP traffic analysis to analyze the top most private and public IP addresses distinctly used in communication as illustrated in Fig. 7. It is discernible that most of the source IP addresses used in TCP and UDP communication are private whilst the majority of prominent destination IP addresses for both protocols are public.

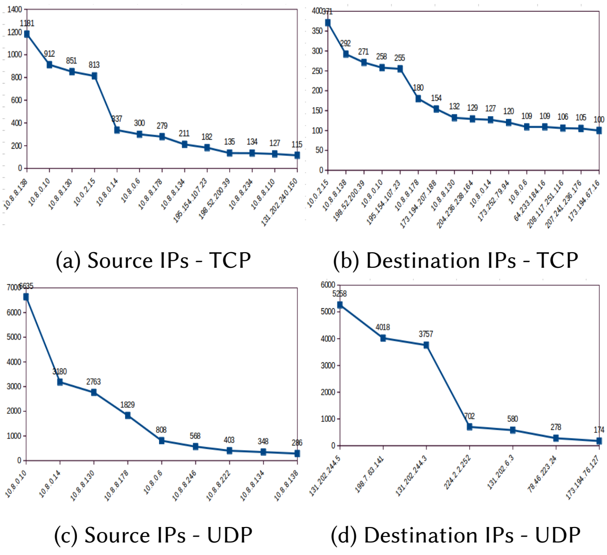


(a) Forward Packets/second (b) Backward Packets/second

(c) TCP: Forward Packets/hr (d) TCP: Backward Packets/hr

(e) UDP: Forward Packets/hr (f) UDP: Backward Packets/hr

Fig. 6. Analysis of TCP and UDP activity traffic



(a) Source IPs - TCP (b) Destination IPs - TCP

(c) Source IPs - UDP (d) Destination IPs - UDP

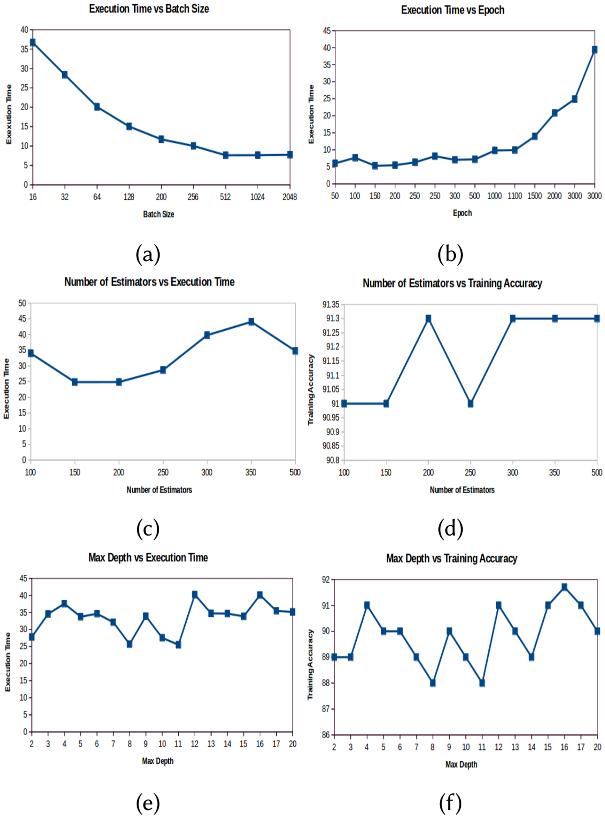
Fig. 7. Analysis of distinct IP-based TCP/UDP traffic

8.6 Hyper-parameter Tuning

Hyper-parameters are tuned to improve the performance of a model. We tested the following hyper-parameters:

- **Execution Time**

Fig. 8 (a and b) clearly show that execution time decreases exponentially with increase in batch size and stables at the end. On the contrary, it remains almost same with increase in epoch value till 1100 but increases exponentially later on.



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 8. Effect of hyper-parameter tuning

- **Number of Estimators in Extra Trees Classifier**

Fig. 8 (c and d) demonstrate that execution time initially decreases as the number of estimators increase and then becomes firm and later on increases rapidly and then finally decreases. Conversely, training accuracy remains consistent around 91% throughout.

- **Max Depth of Extra Trees Classifier**

Fig. 8 (e and f) suggest that there is a fluctuation in execution time with an increase in a maximum depth of forest trees. No steady pattern is observed for execution time. However, Training accuracy varies between 88% and 91.7% with increasing maximum depth values.

9 CONCLUSION AND FUTURE WORKS

We presented DeepImage, a novel approach that classifies and characterizes diverse hidden services and applications in darknet. This is the first time that Tor and VPN traffic is combined to create a real representative of darknet dataset. DeepImage uses feature extraction to select exclusive network header and payload features to create a gray-scale image vector which is fed to a two-dimensional convolutional neural network to distinguish benign traffic from anonymized traffic at layer1 and characterize darknet traffic at layer2. Following conclusions can be drawn from research work in this paper:

- 2D CNN outplays 1D CNN, for the combined dataset, in detecting darknet traffic by an adequate margin. These results encourage identifying diverse protocols and suspicious hidden services.
- Although P2P is one of the most difficult applications to detect because it uses random port numbers and sophisticated port obfuscation techniques to conceal its identity yet DeepImage fruitfully identified 98% of P2P samples in the dataset with least FN rate.
- Hyper-parameter tuning is essential to obtain optimal values of various variables used to select the best features and execute the model to achieve trivial throughput.
- Analysis of backward packets per second and forward packets per second play an imperative role in understanding and characterizing darknet traffic.

Overall, there is a dire need to generate an exhaustive darknet traffic dataset that contains Tor, VPN and Tor over VPN traffic, reports a diverse range of hidden services providing complete interaction, and reveals anonymized IP addresses used in real-time communication with darknet. Producing such a darknet dataset which includes a variety of encrypted traffic and hidden services along with having multi-layered encryption traffic such as Tor over VPN is left as our future work.

ACKNOWLEDGMENTS

We thank the Mitacs Globalink Program for providing the Research Internship (GRI) opportunity to propose deep image learning model that we used in this research paper and Fredrik and Catherine Eaton Visitorship research fund from University of New Brunswick (UNB).

REFERENCES

[1] Waseem Iqbal Abid Khan Jadoon, Muhammad Amjad, Hammad Afzal, and Yawar Abbas Bangash. 2019. Forensic Analysis of Tor Browser: A Case Study for Privacy and Anonymity on the Web. *Forensic Science International* 299 (2019), 59–73.

[2] Masoud Akhoondi, Curtis Yu, and Harsha V. Madhyastha. 2014. LASTor: A Low-Latency AS-Aware Tor Client. *IEEE/ACM Transactions on Networking* 22 (6) (2014), 1742–1752.

[3] Ryoh Akiyoshi, Daisuke Kotani, and Yasuo Okabe. 2018. Detecting Emerging Large-Scale Vulnerability Scanning Activities by Correlating Low-Interaction Honeypots with Darknet. In *42nd IEEE International Conference on Computer Software Applications*. 658–663.

[4] Khaled Al-Naami, Swarup Chandra, Ahmad Mustafa, Latifur Khan, Zhiqiang Lin, Kevin Hamlen, and Bhavani Thuraisingham. 2016. Adaptive Encrypted Traffic Fingerprinting With Bi-Directional Dependence. In *Proceedings of the 32nd Annual Conference on Computer Security Applications*. 177–188.

[5] Mhd Wesam Al-Nabki, Eduardo Fidalgo, Enrique Alegre, and Laura Fernández-Robles. 2019. ToRank: Identifying the most influential suspicious domains in the Tor network. *Expert Systems With Applications* 123 (2019), 212–226.

[6] Riyad Alshammari and A. Nur Zincir-Heywood. 2008. Investigating Two Different Approaches for Encrypted Traffic Classification. In *Sixth Annual Conference on Privacy, Security and Trust*. 156–166.

[7] Riyad Alshammari and A. Nur Zincir-Heywood. 2009. Machine Learning Based Encrypted Traffic Classification: Identifying SSH and Skype. In *Proceedings of the 2009 IEEE Symposium on Computational Intelligence in Security and Defense Applications*. 1–8.

[8] Reyhane Attaria, Lida Abdi, and Sattar Hashemi. 2019. AdaWFPA: Adaptive Online Website Fingerprinting Attack for Tor Anonymous Network: A Stream-wise Paradigm. *Computer Communications* 148 (2019), 74–85.

[9] Carlos Bacquet, Kubra Gumus, Dogukan Tizer, A. Nur Zincir-Heywood, and Malcolm Heywood. 2010. A Comparison of Unsupervised Learning Techniques for Encrypted Traffic Identification. *IJICR* (2010), 1–9.

[10] Sikha Bagui, Xingang Fang, Ezhil Kalaimannan, Subhash C. Bagui, and Joseph Sheehan. 2017. Comparison of machine-learning algorithms for classification of VPN network traffic flow using time-related features. *Journal of Cyber Security Technology* 1 (2) (2017), 108–126.

[11] Tao Ban, Shaoning Pang, Masashi Eto, Daisuke Inoue, Koji Nakao, and Runhe Huang. 2016. Towards Early Detection of Novel Attack Patterns through the Lens of A Large-Scale Darknet. In *IEEE Conferences on Ubiquitous Intelligence Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress.* 341–349.

[12] Laurent Bernaille and Renata Teixeira. 2007. Early Recognition of Encrypted Applications. In *8th Internatinoal Conference on Passive and Active network Measurement.* 165–175.

[13] Laurent Bernaille, Renata Teixeira, and Kave Salamatian. 2006. Early application identification. In *In Proceedings of the Conference on Future Networking Technologies.*

[14] Elias Bou-Harb, Mourad Debbabi, and Chadi Assi. 2014. Cyber scanning: a comprehensive survey. *IEEE Communication Surveys and Tutorials* 16 (3) (2014), 1496–1519.

[15] Julian Caicedo-Muñoz, Agapito Espino, Juan Corrales, and Alvaro Rendón. 2018. QoS-Classifier for VPN and Non-VPN traffic based on time-related features. *Computer Networks* 144 (2018), 271–279.

[16] Milan Cermak, Tomas Jirsik, Petr Velan, Jana Komarkova, Stanislav Spacek, Martin Drasar, and Tomas Plesnik. 2018. Towards Provable Network Traffic Measurement and Analysis via Semi-Labeled Trace Datasets. In *2018 Network Traffic Measurement and Analysis Conference (TMA).* 1–8.

[17] Kevin Chen, Jennifer Tu, and Alex Vandiver. 2004. Analyzing Network Traffic from a Class B Darknet. *MIT* (2004).

[18] CICFlowMeter. Access February 2017. Ethernet Traffic Flow Meter. https://github.com/ahlashkari/CICFlowMeter.

[19] Fangzhou Dong, Shaoxian Yuan, Haoran Ou, and Liang Liu. 2018. New Cyber Threat Discovery from Darknet Marketplaces. In *IEEE Conference on Big Data and Analytics.* 62–67.

[20] Claude Fachkha, Elias Bou-Harb, and Mourad Debbabi. 2015. Inferring distributed reflection denial of service attacks from darknet. *Computer Communications* 62 (2015), 59–71.

[21] Claude Fachkha, Elias Bou-Harb, Anastasis Keliris, Nasir Memon, and Mustaque Ahamad. 2017. Internet-scale probing of cps: Inference, characterization and orchestration analysis. *In Proceedings of Network and Distributed System Security Symposium* 17 (2017), 100–113.

[22] Claude Fachkha and Mourad Debbabi. 2016. Darknet as a Source of Cyber Intelligence: Survey, Taxonomy, and Characterization. *IEEE Communications Surveys & Tutorials* 18 (2) (2016), 1197–1227.

[23] Ed Wilson Tavares Ferreira and Ailton Akira Shinoda. 2016. The Development and Evaluation of a Dataset for Testing of IDS for Wireless Networks. *IEEE Latin America Transactions* 14, 1 (2016), 404–410.

[24] Eduardo Fidalgo, Enrique Alegre, Laura Fernández-Robles, and Víctor González-Castro. 2019. Classifying suspicious content in tor darknet through Semantic Attention Keypoint Filtering. *Digital Investigation* 30 (2019), 12–22.

[25] Falguni Gadhia, Jangwon Choi, Buseung Cho, and Jungsuk Song. 2015. Comparative analysis of darknet traffic characteristics between darknet sensors. In *International Conference on Advanced Communication Technology.* 59–64.

[26] Gephi. [n.d.]. Gephi, the leading visualization and exploration software for all kinds of graphs and networks. https://gephi.org/. , Accessed February 2020 pages.

[27] Gerard Draper Gil, Arash H. Lashkari, Mohammad Saiful Islam Mamun, and Ali A. Ghorbani. 2016. Characterization of Encrypted and VPN Traffic Using Time-Related Features. In *In Proceedings of the 2nd International Conference on Information Systems Security and Privacy.* 407–414.

[28] Ramzi A. Haraty and Bassam Zantout. 2014. The TOR Data Communication System. *Journal of Communications and Networks* 16 (4) (2014), 415–420.

[29] Naoki Hashimoto, Seiichi Ozawa, Tao Ban, Junji Nakazato, and Jumpei Shimamura. 2018. A Darknet Traffic Analysis for IoT Malwares Using Association Rule Learning. *Conference on Big Data and Deep Learning, Procedia Computer Science* 144 (2018), 118–123.

[30] Gaofeng He, Ming Yang, Junzhou Luo, and Xiaodan Gu. 2014. Inferring Application Type Information from Tor Encrypted Traffic. In *Second International Conference on Advanced Cloud and Big Data.* 220–227.

[31] C. Rosenburg J. Early, C. Brodley. 2003. Behavioral authentication of server flows. In *In Proceedings of the 19th Annual Computer Security Applications Conference.*

[32] Husam Al Jawaheri, Mashael Al Sabah, Yazan Boshmaf, and Aiman Erbad. 2020. Deanonymizing Tor hidden service users through Bitcoin transactions analysis. *Computers & Security* 89 (2020).

[33] Kota Kanemura, Kentaroh Toyoda, and Tomoaki Ohtsuki. 2019. Identification of Darknet Markets' Bitcoin Addresses by Voting Per-address Classification Results. In *IEEE International Conference on Blockchain and Cryptocurrency.* 154–158.

[34] Doron Kolton, Adi Stav, Asaf Wexler, Ariel Ernesto Frydman, and Yoram Zahavi. 2011. System to enable detecting attacks within encrypted traffic. *United States Patent* No. US 7,895,652 B2 (2011), 1–9.

[35] Yuichi Kumano, Shingo Ata, Nobuyuki Nakamura, Yoshihiro Nakahira, and Ikuo Oka. 2014. Towards Real-time Processing for Application Identification of Encrypted Traffic. In *International Conference on Computing, Networking and Communications, Communication QoS and System Modeling Symposium.* 136–140.

[36] Arash Habibi Lashkari, Gerard Draper-Gil, Mamun Seiful Islam, and Ali Ghorbani. 2017. Characterization of Tor Traffic Using Time Based Features. In *In the proceeding of the 3rd International Conference on Information System Security and Privacy, SCITEPRESS.* 253–262.

[37] Zhen Ling, Junzhou Luo, Kui Wu, Wei Yu, and Xinwen Fu. 2015. TorWard: Discovery, Blocking, and Traceback of Malicious Traffic Over Tor. *IEEE Transactions on Information Forensics and Security* 10 (12) (2015), 2515–2530.

[38] Zhen Ling, Junzhou Luo, Wei Yu, Xinwen Fu, Weijia Jia, and Wei Zhao. 2013. Protocol-level attacks against Tor. *Computer Networks* 57 (2013), 869–886.

[39] Jun Liu and Kensuke Fukuda. 2014. Towards a Taxonomy of Darknet Traffic. In *International Wireless Communications and Mobile Computing Conference.* 37–43.

[40] Tomáš Liška, Tomáš Sochor, and Hana Sochorová. 2011. Comparison between normal and TOR-Anonymized Web Client Traffic. *Procedia Computer Science* 3 (2011), 888–892.

[41] Mohammad Lotfollahi, Ramin Shirali Hossein Zade, Mahdi Jafari Siavoshani, and Mohammdsadegh Saberian. 2020. Deep packet: a novel approach for encrypted traffic classification using deep learning. *Soft Computing* 24 (2020), 1999–2012.

[42] Shane Miller, Kevin Curran, and Tom Lunney. 2018. Multilayer Perceptron Neural Network for Detection of Encrypted VPN Network Traffic. In *IEEE International Conference on Cyber Situational Awareness, Data Analytics and Assessment.*

[43] Tomáš Minárik and Anna-Maria Osula. 2016. Tor does not stink: Use and abuse of the Tor anonymity network from the perspective of law. *Computer Law Security Review* 32 (2016), 111–127.

[44] Mihnea Mirea, Victoria Wang, and Jeyong Jung. 2019. The not so dark side of the darknet: a qualitative study. *Security Journal* 32 (2019), 102–118.

[45] Antonio Montieri, Domenico Ciuonzo, Giuseppe Aceto, and Antonio Pescapé. 2018. Anonymity Services Tor, I2P, JonDonym: Classifying in the Dark (Web). *IEEE Transactions on Dependable and Secure Computing* (2018), 1–14.

[46] Antonio Montieri, Domenico Ciuonzo, Giampaolo Bovenzi, Valerio Persico, and Antonio Pescapé. 2019. A Dive into the Dark Web: Hierarchical Traffic Classification of Anonymity Tools. *IEEE Transactions on Network Science and Engineering* (2019).

[47] David Moore, Geoffrey M. Voelker, and Stefan Savage. 2006. Inferring internet denial-of-service activity. *ACM Transactions on Computer Systems* 24 (2) (2006), 115–139.

[48] Amarnath Mullick, Shashi Nanjundaswamy, Charu Venkatraman, Junxiao He, James Harris, and Ajay Soni. 2014. Systems and methods for application based interception of SSL/VPN traffic. *Journal of Network and Computer Applications* No. US 8,869,262 B2 (2014), 1–10.

[49] Thomas D. Nadeau, Sumit Mukhopadhyay, Stephen Paul Elias, and Adrien Michael Grise. 2010. Methods and apparatus providing VPN traffic matrix construction. *United States Patent* No. US 7,839,847 B2 (2010), 1–8.

[50] Hironori Nishikaze, Seiichi Ozawa, Jun Kitazono, Tao Ban, Junji Nakazato, and Jumpei Shimamura. 2015. Large-Scale Monitoring for Cyber Attacks by Using Cluster Information on Darknet Traffic Features. *2015 INNS Conference on Big Data, Procedia Computer Science* 53 (2015), 175–182.

[51] Gareth Owenson, Sarah Cortes, and Andrew Lewman. 2018. The darknet's smaller than we thought: The life cycle of Tor Hidden Services. *Digital Investigation* 27 (2018), 17–22.

[52] Kunal Patel, Yixin Sun, Puneet Gupta, Vinod Arjun, and David McGrew. 2014. Techniques To Classify Virtual Private Network Traffic Based On Identity. *United States Patent* No. US 8,909,918 B2 (2014), 1–6.

[53] Morteza Safaei Pour and Elias Bou-Harb. 2019. Theoretic derivations of scan detection operating on darknet traffic. *Computer Communications* 147 (2019), 111–121.

[54] Shahbaz Rezaei and Xin Liu. 2019. Deep Learning for Encrypted Traffic Classification: An Overview. *Data Science And Artificial Intelligence For Communications* (2019), 76–81.

[55] A. Nur Zincir-Heywood Riyad Alshammari. 2011. Can encrypted traffic be identified without port numbers, IP addresses and payload inspection? *Computer Networks* 55 (2011), 1326–1350.

[56] Saad Saleh, Junaid Qadir, and Muhammad U. Ilyas. 2018. Shedding Light on the Dark Corners of the Internet: A Survey of Tor Research. *Journal of Network and Computer Applications* 114 (2018), 1–28.

[57] Scikit. [n.d.]. Feature importance with forests of trees. https://scikit-learn.org/stable/auto_examples/ensemble/plot_forest_importances.html#sphx-glr-auto-examples-ensemble-plot-forest-\importances-py. , Accessed February 2020 pages.

[58] Scikit. [n.d.]. scikit-learn: Machine Learning in Python. https://scikit-learn.org/stable/. , Accessed February 2020 pages.

[59] Khalid Shahbar and A. Nur Zincir-Heywood. 2014. Benchmarking Two Techniques for Tor Classification: Flow Level and Circuit Level Classification. In *IEEE Symposium on Computational Intelligence in Cyber Security.*

[60] Iman Sharafaldin, Arash H. Lashkari, and Ali A. Ghorbani. 2018. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. In *Proceedings of the 4th International Conference on Information Systems Security and Privacy*. 108–116.

[61] Anish Singh Shekhawat, Fabio Di Troia, and MarkStamp. 2019. Feature analysis of encrypted malicious traffic. *Expert Systems With Applications* 125 (2019), 130–141.

[62] Meng Shen, Mingwei Wei, Liehuang Zhu, and Mingzhong Wang. 2017. Classification of Encrypted Traffic With Second-Order Markov Chains and Application Attribute Bigrams. *IEEE Transactions On Information Forensics And Security* 12 (8) (2017), 1830–1843.

[63] Justine Sherry, Chang Lan, Raluca Ada Popa, and Sylvia Ratnasamy. 2015. BlindBox: Deep Packet Inspection over Encrypted Traffic. *ACM SIGCOMM* (2015), 213–226.

[64] Francesca Soro, Idilio Drago, Martino Trevisan, Marco Mellia, João Ceron, and José J. Santanna. 2019. Are Darknets All The Same? On Darknet Visibility for Security Monitoring. In *IEEE International Symposium on Local and Metropolitan Area Networks*. 1–6.

[65] Martijn Spitters, Stefan Verbruggen, and Mark van Staalduinen. 2014. Towards a Comprehensive Insight into the Thematic Organization of the Tor Hidden Services. In *IEEE Joint Intelligence and Security Informatics Conference*.

[66] Guang-Lu Sun, Yibo Xue, Yingfei Dong, Dongsheng Wang, and Chenglong Li. 2010. An Novel Hybrid Method for Effectively Classifying Encrypted Traffic. In *Proceedings of the Global Communications Conference*. 1–5.

[67] Tor. [n.d.]. Tor Metrics. https://metrics.torproject.org. , Accessed March 2020 pages.

[68] William Turkett, Andrew Karode, and Errin Fulp. 2008. In-the-Dark Network Traffic Classification Using Support Vector Machines. In *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence*. 1745–1750.

[69] Meiqi Wang, Xuebin Wang, Jinqiao Shi, Qingfeng Tan, Yue Gao, Muqian Chen, and Xiaoming Jiang. 2018. Who are in the Darknet? Measurement and Analysis of Darknet Person Attributes. In *IEEE Third International Conference on Data Science in Cyberspace*. 948–955.

[70] Wei Wang, Ming Zhu, Jinlin Wang, Xuewen Zeng, and Zhongzhen Yang. 2017. End-to-end Encrypted Traffic Classification with One-dimensional Convolution Neural Networks. In *IEEE International Conference on Intelligence and Security Informatics*. 43–48.

[71] Xiaogang Wang, Junzhou Luo, Ming Yang, and Zhen Ling. 2011. A potential HTTP-based application-level attack against Tor. *Future Generation Computer Systems* 27 (2011), 67–77.

[72] Charles Wright, Fabian Monrose, and Gerald Masson. 2006. On inferring application protocol behaviors in encrypted network traffic. *Journal of Machine Learning Research* 7 (2006).

[73] Roni Bar Yanai, Michael Langberg, David Peleg, and Liam Roditty. 2011. Real-time Encrypted Traffic Identification using Machine Learning. *Journal of Software* 6 (6) (2011), 1009–1016.

[74] Qing Yanga, Paolo Gastib, Kiran Balaganib, Yantao Lic, and Gang Zhou. 2018. USB side-channel attack on Tor. *Computer Networks* 141 (2018), 57–66.

[75] Han Zhang, Christos Papadopoulos, and Dan Massey. 2013. Detecting Encrypted Botnet Traffic. *16th IEEE Global Internet Symposium* (2013), 3453–3458.