SCTE·ISBE CABLE-TEC EXPO 2019 NEW ORLEANS, LA SEPT. 30-OCT. 3

2019 Fall Technical Forum SCTE·ISBE • NCTA • CABLELABS

Detecting Video Piracy with Machine Learning

A Technical Paper prepared for SCTE•ISBE by

Matthew Tooley
Vice President of Broadband Technology
NCTA – The Internet & Television Association
Washington, DC
(202) 222-2479
mtooley@ncta.com

Thomas Belford
Software Engineer – Technology Department
NCTA – The Internet & Television Assocation
Washington, DC
thomasbelford32@gmail.com

Table of Contents

List of Figures

List of Tables

Introduction

The broad adoption of broadband internet and growth in average internet speed [1] has fueled the streaming video industry. In turn, the growth and popularity of streaming video has also fueled the growth of video piracy [2].

Video piracy is a form of copyright infringement and refers to the use of works protected by copyright law without permission for usage where such permission is required. There are two primary forms of video piracy. The first form commonly referred to as "video-on-demand" (VOD) uses a file sharing distribution model and is commonly used by applications such as Kodi, Titanium TV, TVZion and BitTorrent based applications [3].

The popularity of streaming video has resulted in the creation of illegal virtual cable operators selling subscription based over-the-top IPTV, complete with electronic programming guides, that stream multiple channels of linear video. This second form is known as "pirated linear streaming".

Pirated linear streaming is a business threat to the pay-TV industry as the pirated linear streaming product is a good substitute for legitimate pay-TV services. For the pay-TV industry, one of the issues is understanding the true scope of the problem. There are some industry reports [4] that estimate that 5.5% of North American households are accessing pirated content. The pay-TV industry has been trying to better quantify the problem, as part of determining what actions to take to mitigate it.

To truly understand the scope and scale of video piracy, operators need to measure the volume, frequency and scope of traffic on their networks that is associated with pirated linear streams. Pirated streams use the same technologies and streaming protocols (HLS and MPEG/DASH) as legal linear streams making it difficult to distinguish the two without the use of deep packet inspection (DPI). Even with DPI, it is still difficult due to multi-tenant hosts, content delivery networks, multiple IP addresses being associated with the content sources, and the diverse demographics across the footprint of the network.

Due to a number of reasons including cost and privacy concerns, operators typically have only equipped a small portion (e.g. < 10%) of their network with DPI, if at all. In addition, collecting video piracy data using DPI from a small number of points on the network can lead to a selection bias due to the demographic makeup of the network footprint.

To effectively measure video piracy on broadband networks requires something other than DPI. An approach using available IPFIX/NetFlow data, which is embedded in most carrier-grade routers and switches, provides a cost-effective approach to measuring traffic across an entire network.

In 2016 Cisco [5] showed that by using IP flow data fields it was possible to create a feature set for machine learning that used an L1-logistic regression model with an accuracy of 99.978% at 0.00% false discovery rate (FDR) to identify malware – encrypted and non-encrypted. In 2018, Cisco [6] introduced an enhanced version of NetFlow, Encrypted Traffic Analytics (ETA), that included these additional IP flow data fields to a number of its products as part of a cybersecurity solution and open-sourced the code[1] that captures, extracted, and analyzes network flow data and interflow data that includes the additional IP flow data fields.

In this paper, we look at applying a similar supervised machine learning process using IP flow data to assess the viability of using machine learning and IP flow data to detect pirated linear streaming traffic on broadband networks.

Content

1. Deeper Look at Video Piracy Traffic

Building upon the work by Anderson and McGrew, which showed machine learnings capability to detect the unique signatures of malware, we began by inspecting both legal and illegal IP video streams. Using machine learning, we tried to identify the features that have the most discriminatory power, knowing that these features will be able to uniquely identify pirated from non-pirated traffic.

In today's video pirate ecosystem, most providers are using a client/server software pair from a small number of providers.[2] This fact coupled with the long-tailed nature of linear streaming video makes it possible to identify a set of features with strong discriminatory power.

For this study we captured packet capture files from a set of known pirate subscription sites as listed in Table 1.

Table 1 Pirate IPTV Providers Analyzed

Pirate IPTV Provider	Homepage URL
IPTV Shop	https://iptv.shop
Excursion TV – Premium USA IPTV	https://www.excursion-tv.com
IPTV Choice	https://iptvchoice.com
Easy Expat IPTV	https://easyexpatiptv.org
Gears TV HD	https://www.gearstvhd.com
Necro IPTV: IPTV	https://necroiptv.com
Nitro IPTV	https://www.iptvnitro.com
SoftIPTV	https://www.softiptv.com

For all the pirate subscription services a common flow session is as shown in Figure 1.

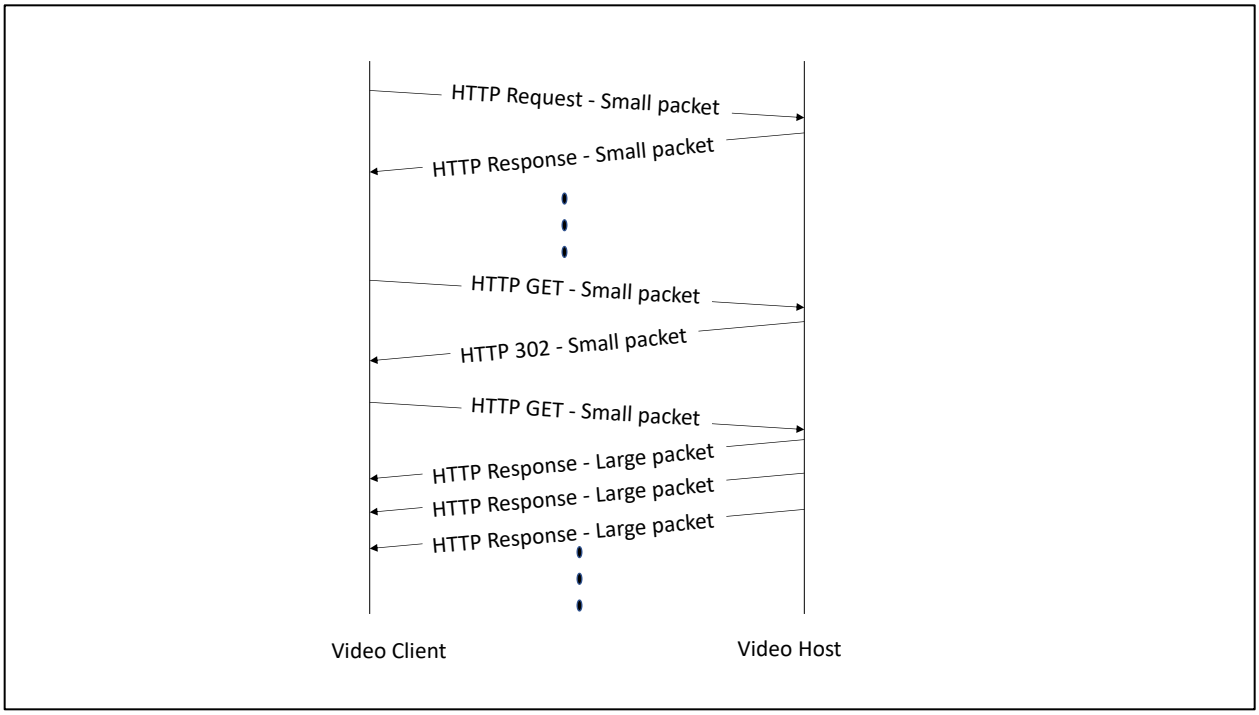

Figure 1 Common Pirated Video Flow Sequence

Video streaming typically includes one or more long-tailed flows, that are initially preceded by a series of small HTTP transactions or short-tailed flows where the video client is logging into the back-end, followed by a the video client selecting a channel and the backend server sending an HTTP redirect to redirect the video client to the location of the video which may either be on a dedicated server or on a content delivery network (CDN).

The linear streamed video is delivered using either the HTTP Live Streaming (HLS) [7] protocol or the MPEG DASH [8].

For each of the packet captures, we looked at the flows. A flow is defined as the traffic between the 4-tuple (source & destination address, source & destination ports) and their data features. We compared the data features of the pirate linear streaming traffic to the data features of other forms of streaming video and benign internet traffic to evaluate which should have the most discriminatory power when used in a machine learning model. The data features we studied are listed in Table 2.

Table 2 IP Flow Data Features of Interest

Data Feature	Description
Packet sizes per flow	The size of each packet in the flow
Number of packets per flow	The number of packets in the flows
Number of bytes per flow	Total number of payload bytes in the flow

Flow duration	Time in seconds of the flow from the TCP SYN to the TCP FIN
Inter-packet time per flow	The number of milliseconds between each packet in the flow
Source Port	The source port of the flow
Destination Port	The destination port of the flow
Byte Distribution	The frequency of occurrence of the byte values in the first "n" packet payload of the first packet of the flow

We first looked at how the pirated linear streaming services compared to some of the more popular over-the-top (OTT) video services – Youtube, Netflix and Twitch. Figure 2 shows the histograms for the features listed in Table 2, and shows that the video piracy has a number of features that have distributions that differ from Youtube. Figure 3 shows a larger version of the histogram for the packet sizes in the IP flow. Even though both are forms of long-tail video, as can be seen in the figures, there are still distinct data features that emerge. The same is true when video piracy is compared to other forms OTT video.

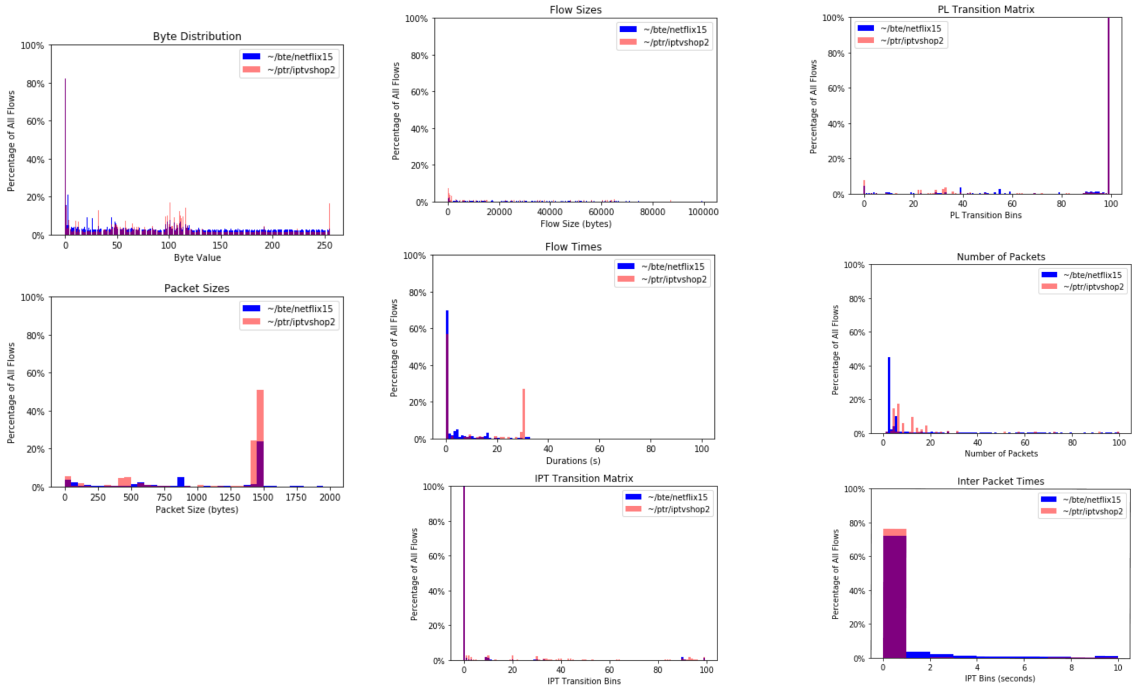

Figure 2 Example Netflix(Blue) vs Pirate IPTV (Red) Feature Comparison

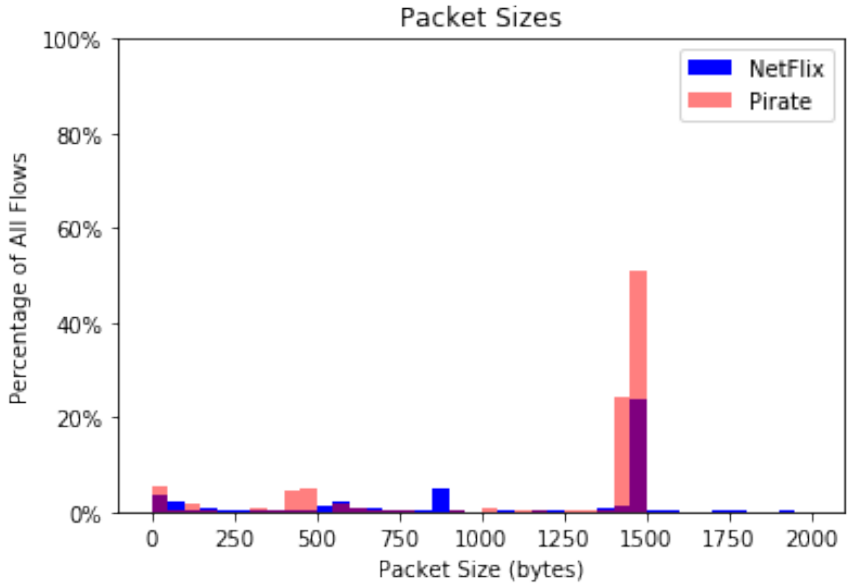

Figure 3 Packet Sizes per Flow Comparison of Netflix(Blue) vs Video Piracy(Red)

Next, we compared the pirate video traffic to a collection of internet traffic. The internet traffic collection contains captures for multiple forms of short-tail internet traffic including web browsing, webmail, mobile phone, and cloud storage. As shown in the Figure 4, just as when the pirate video traffic is compared to the OTT video, the pirate video traffic has unique characteristics that make its data features unique when compared to the internet traffic collection.

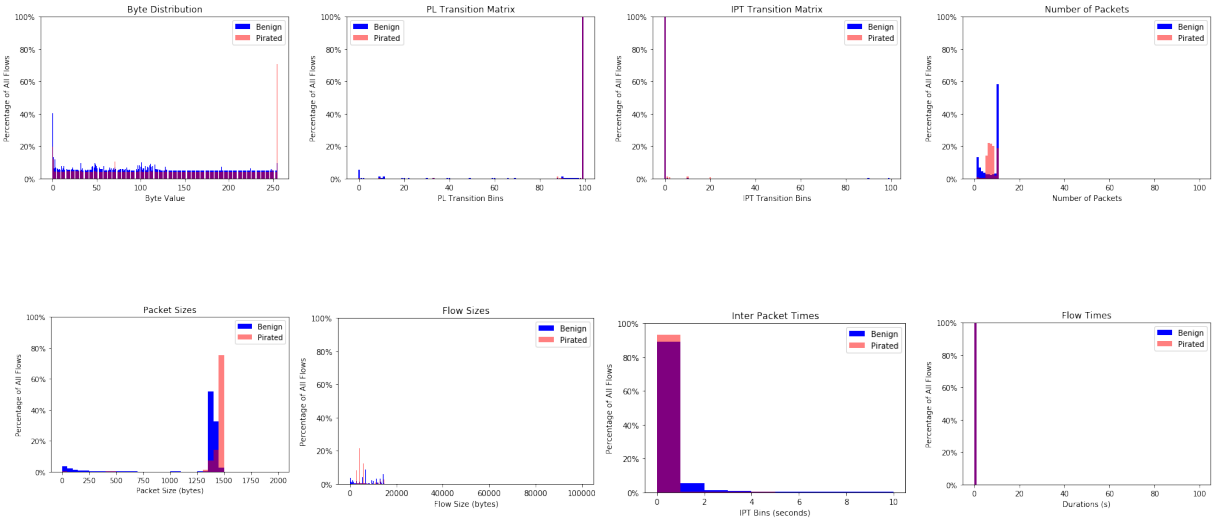

Figure 4 Feature Comparison of Pirate (Red) vs Internet (Blue)

2. Machine Learning

2.1. Overview

Machine learning is a subset of Artificial Intelligence (AI) and uses algorithms to discover patterns in data and constructs mathematical models using these discoveries. The models can then be used to make predictions on future data. Using machine learning to perform traffic classification is not new [9]; however, the use of IP flow data and NetFlow as the transport mechanism and synthesizing features from the flow data is new. Machine learning can either be supervised or unsupervised. Supervised machine learning trains or teaches the machine using data that is labeled with the correct answer, e.g. pirated or not, while unsupervised machine learning trains the machine using information that is neither classified nor labeled and allows the algorithm to act on the information without guidance.

Because of this, we embraced supervised machine learning as the best way to use previously observed video piracy to detect video piracy. Further, a supervised machine learning classifier provides the most direct way to build a detector, and it can also provide a probability estimate.

Machine learning makes use of the following terminology:

- **Machine Learning Algorithm** – Machine learning algorithms build the mathmetical model based on the 'training data". There are a number of machine learning algorithms, including Logistic Regression, Decision Trees, and Random Forest.
- **Model** – A model is a specific representation learned from data by applying some machine learning algorithm.
- **Feature** – A feature is an individual measurable property of data. A set of numeric features can be conveniently described by a **feature vector**. Feature vectors are fed as input to the model.
- **Feature Extraction** – Feature extraction refers to the method and process of constructing data from the intial raw set of data to a more maneageable group for processing.
- **Label** – A label is the value to be predicted by the model.
- **Training** – The process of creating a model or classifier using a machine learning algorithm.
- **Prediction** – Predicted output from a set of inputs

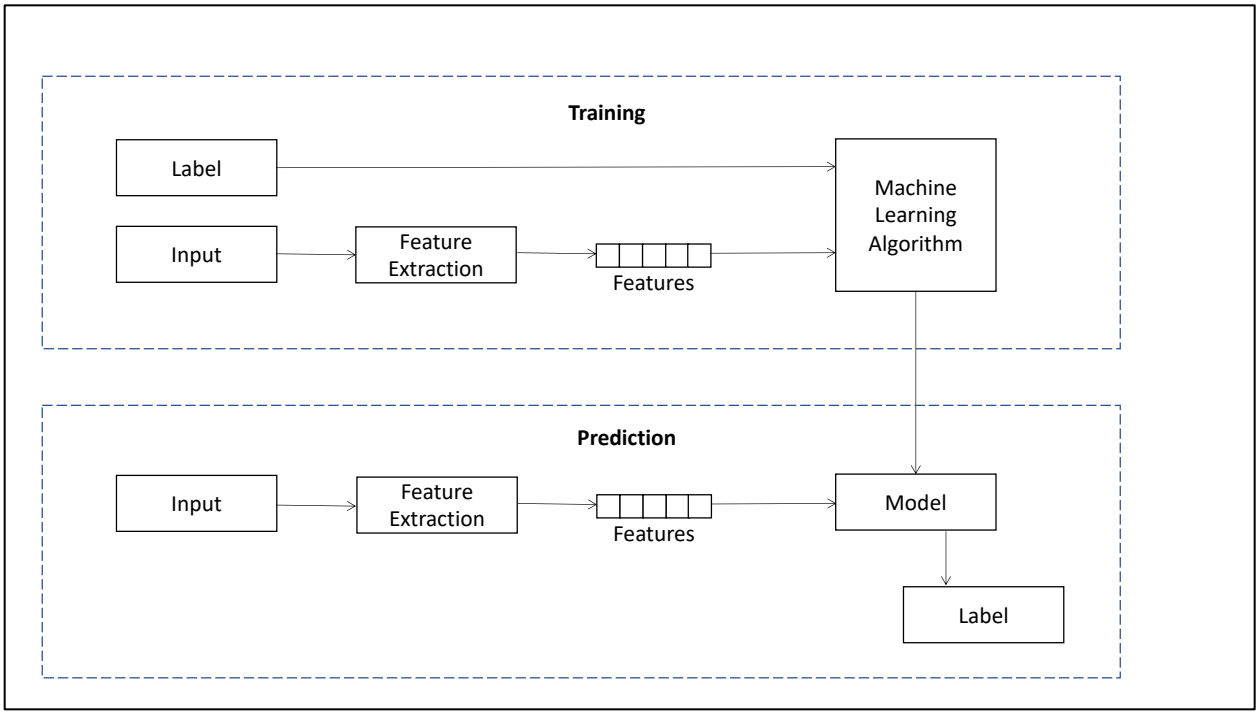

Figure 5 Concept of Machine Learning

2.2. Machine Learning Algorithm

Detecting video piracy is a classification problem, and we are treating it as a binary classification problem as we want to predict if the input is pirated video or not. We looked at three different machine learning algorithms to assess which model perfomed the best for classifying video piracy using flow data.

- **Logistic Regression** – Logistic Regression is a predictive analysis classification algorithm and based on the concept of probability. It is used to assign observations to a discrete set of classes and transforms it output using a logistic sigmoid function to return a probability value.
- **Decision Tree –** Decision tree builds classification into a form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. Decision tree uses entropy and information gain to construct a decision tree.
- **Random Forest Classification –** Random Forest Classifier is an ensemble-based algorithm which is comprised of n collections of de-correlated decision trees [10]. Random Forest uses multiple trees to compute majority votes in the terminal leaf nodes when making a prediction.

We also briefly looked into Neural Networks and Support Vector Machines, however, the size of our data and feature sets were not optimal for these models, so we decided to disclude them from the study. Of the three we studied in depth, Logistic Regression is computationally the most efficient, while Decision Tree and Random Forest are often more accurate while being computationally more intense.

2.3. Data Features

Feature selection is most important step in machine learning. For the feature selection, we looked at the data features available in the flow data as shown in Table 3. The flow data we looked at could either be

extracted from IPFIX/NetFlow v9 or from NetFlow enriched with a set of user defined fields that are available in a proprietary form of NetFlow from Cisco or from open source flow data feature extractor called "Joy".

Table 3 Features

Field	Source	Description
IP Protocol	NetFlow v9	TCP or UDP
Source Port	NetFlow v9	Source port in the IP header
Destination Port	NetFlow v9	Destination port in the IP header
Flow Length, Bytes	NetFlow v9	Number of bytes for the TCP connection
Flow Duration, Seconds	NetFlow v9	Duration of the flow from TCP SYNC to TCP FIN
Packets/Flow	Enhanced NetFlow	Number of the packets for the IP flow
Sequence of Packet Lengths and Times (SPLT)	Enhanced NetFlow	An array of LENGTH values followed by an array of INTER-ARRIVAL TIME values describing the first N packets of a flow that carry application payload. Each LENGTH is encoded as a 16-bit integer to form a 20-byte array. Immediately following this, each INTER-ARRIVAL TIME is encoded as a 16-bit integer to form another 20-byte array.
Byte Distribution	Enhanced NetFlow	A histogram giving the frequency of occurrence for each byte value or (range of values) in the first N bytes of application payload for a flow. Each "frequency of occurrence" is represented as a 16-bit integer.
Initial Packet Data (IPD)	Enhanced NetFlow	The content of the first packet of this flow that contains actual payload data, starting at the beginning of the IP header.

Joy[3] was used to extract the flow data from either from packet captures or collected live as flow data records (i.e. NetFlow) into a JSON format as part of the feature extraction. Joy models **packet lengths** and **packet inter-arrival times** as Markov chains (a sequence of possible events) and excludes TCP retransmissions. The packet length is the payload length of the packet. Inter-arrival times have milli-second resolution.

For both the lengths and times, the values are discretized into equally sized bins. The length data Markov chain has 10 bins of 150 bytes. The timing data Markov chain uses 50 millisecond bins and 10 bins for

100 total features. The Markov chains are transformed into their transition probabilities are used as the features.

Joy extracts the **byte distribution data** from the flow data and generates a 256-byte distribution probability from the 256-byte distribution array for the feature.

Additional **IP meta-data** is also included in the features. The IP meta-data includes source port, destination port, number of packets in, number of packets out, number of bytes in, number of bytes out, and flow duration.

2.4. Data Sets

To analyze the models, we created four data sets. A pair of training data sets and a pair of test sets were created using a packet capture program. To train the model on the pirate video traffic that we wanted to identify, we used a packet capture with a set of long-tail flows between two video pirate hosts. To train the model on traffic we did not want to classify or label as benign traffic, we used a packet capture file with a mix of internet browsing, legal OTT video, email, and other enterprise traffic. To test the model we create a second, unique pair of packet capture files. For the benign traffic we simply used a packet capture from an enterprise network that was known not to include video piracy. For the video piracy test file, we performed a packet capture that included a video piracy session between the pirate video client and the pirate video server.

Table 4 - Data Sets

Data Set	Number of Flows	Description
Benign Training	54,726	Enterprise traffic, cloudfront, twitchtv, webmail, web browsing, akamai CDN traffic
Benign Test	14,768	Google search, Netflix, TwitchTV15
Piracy Training	94,742	Expat IPTV channel 3; Gears IPTV HBO; IPTVChoice NBC, PrimeAtlantic, ESPN; IPTVShop 3 &4; Unlock
Piracy Test	3,611	Gears IPTV, Necro, and Vaderstreams

2.5. Machine Learning Model Performance

We used the four data sets to calculate the classification metrics for different combinations of machine learning algorithms and feature sets. To evaluate the performance of the machine learning models we used six standard machine learning metrics. Our goal was to minimize false positives while maximizing accuracy and precision since it would be better for a cabel operator to miss a few pirated flows as opposed to classifying all pirated flows while also misclassifying a lot of benign flows as pirated.

2.5.1. Machine Learning Metrics

For each algorithm we used the classification metrics – accuracy, true and false positive, precision, F1, and Log Loss – to determine which algorithm worked best for classifying video piracy.

2.5.1.1. Accuracy

Accuracy is the ratio of correct predictions to the total number of input samples.

$$Accuracy = \frac{Number\ Correct\ Pirated\ \&\ Benign\ Predictions}{Total\ Number\ of\ Predictions\ Madel}$$

Accuracy works well if there are an equal number of samples belonging to each class.

2.5.1.2. Precision

Precision is the number of correct positive results divided by the number of positive results predicted by the classifier and is intuitively the ability of the classifier not to label as positive a sample that is negative.

$$Precision = \frac{True\ Pirated}{Actual\ Results} = \frac{True\ Pirated}{True\ Pirated + False\ Pirated}$$

2.5.1.3. True Positive Rate (Sensitivity)

True Positive Rate, also known as Recall, corresponds to the proportion of positive data points that are correctly considered as positive, with respect to all data points. True Positive Rate provides a measure of how sensitive the classifier is, and how well it is at not missing actual positives.

$$True\ Positive\ Rate\ or\ Recall = \frac{True\ Pirated}{Predicted\ Results} = \frac{True\ Pirated}{False\ Benign + True\ Pirated}$$

where,

True Positive is the number of cases in which the model predicted YES and the actual output was also YES.

False Negative is the number of cases in which the model predicted NO and the output was YES.

2.5.1.4. False Positive Rate (Fall-Out)

False Positive Rate or Fall-Out corresponds to the proportion of negative data points that are mistakenly considered as positive, with respect to all negative data points. False Positive provides a measure of the classifier's probability of falsely rejecting the null hypothesis.

$$False\ Positive\ Rate = \frac{False\ Pirated}{False\ Pirated + True\ Benign}$$

where,

False Positive is the number of cases in which the model predicted YES and the acutal output was NO.

True Negative is the number of cases in which the model predicted NO and the actual output was NO.

2.5.1.5. F1 Score

F1 is used to measure a test's accuracy and is the harmonic average between precision and recall. It tells how precise the models classifier is, as well as how robust it is. The greater the F1 score the better the performance of the model.

$$F1 = 2 * \frac{1}{\frac{1}{precision} + \frac{1}{recall}}$$

2.5.1.6. Logarithmic Loss

Logarithmic Loss, works by penalizing false classifications.

$$Logarithmic\ Loss = -\frac{1}{N} * \sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij} + Log(p_{ij})$$

Where,

y_{ij}, indicates whether the sample I belongs to class j or not

p_{ij}, indicates the probability of sample I belonging to class j.

2.5.2. Machine Learning Model Peformance Results

We analyzed the three machine learning algorithms and the generatred models using three different combinations of feature sets to determine the model with the best performance. We used the same training and test data sets as described in Table 4 with each. Table 5 shows the results. As shown in table 6, it can be seen that the Random Forest using the largest feature set has the best overall performance. The Random Forest with the full feature set has both a high accuracy and low false positive. The results also show that Random Forest with just meta-data will still identify a large percentage of the video piracy, but may have a high false positive rate that needs to be filtered out with additional post processing.

Table 5 Machine Learning Algorithms Performance Results

Feature Set – Flow Meta Data						
	Accuracy	Precision	F1 Score	Log Loss	True Positive Rate	False Positive Rate
Logistic Regression	21%	20%	33%	27	98.4%	80%
Decision Trees	82%	52%	67%	6.36	98.8%	48%
Random Forest	81%	51%	67%	6.6	98.8%	49%

Feature Set – Flow Meta Data + Byte Distribution						
	Accuracy	Precision	F1 Score	Log Loss	True Positive Rate	False Positive Rate
Logistic Regression	21%	20%	33%	27	98.4%	80%
Decision Trees	99%	97%	97%	.45	99%	3%
Random Forest	99.5%	99.5%	98%	.15	99.5%	0.5%

Feature Set – Flow Meta Data + Byte Distribution + Packet Lengths + Packet Timing						
	Accuracy	Precision	F1 Score	Log Loss	True Positive Rate	False Positive Rate
Logistic Regression	21%	19%	33%	27	98.4%	80%
Decision Trees	96%	96%	88%	1.5	96%	3.7%
Random Forest	97%	99.8%	93%	.90	97%	0.19%

2.6. Machine Learning with NetFlow

On large networks, such as those operated by cable operators, NetFlow is configured to operate in sampled mode. Sampled NetFlow means the NetFlow exporter (i.e. router or switch) is sampling every n^{th} packet to update the flow state tables. Large operators configure n to be anywhere from 1,000 to 4,000. In other words, the NetFlow exporter is sampling every $1{,}000^{th}$ packet or every $4{,}000^{th}$ packet.

As the goal here was to determine the feasibility of using machine learning with flow data for piracy detection on large networks, we needed to analyze the impact of using sampled NetFlow for the flow data and feature extraction; however, due to time constraints, we weren't able to calculate the performance analysis with sampled NetFlow.

Despite the fact that unsampled NetFlow provides only the meta-data features, performance metrics indicate that by using the meta-data the model can identify and classify video piracy, but with a high false positive rate. We expect that reducing the flow data rate from every packet to every n^{th} (i.e. $1{,}000^{th}$ or $4{,}000^{th}$) will further degrade performance, causing the feature extractor to take longer to assemble a flow with enough information associated with it to be positively classified.

3. Case Study

We performed two case studies to evaluate how well the machine learning classifier works in the real world. The first case study used data collected from an enterprise with about 100 employees. The second case study used NetFlow data from two different cable operators.

3.1. Implementation

For the case studies, we implemented the system shown in Figure 6 Machine Learning Video Piracy Detection System.

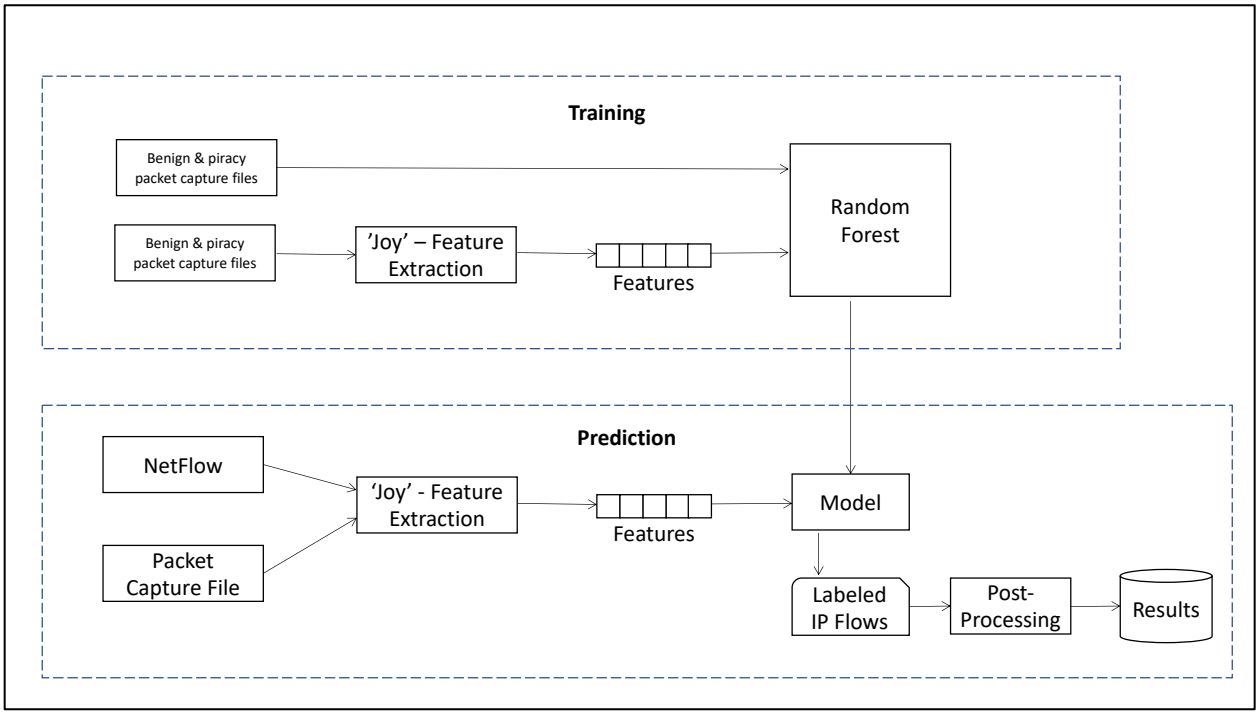

Figure 6 Machine Learning Video Piracy Detection System

For the system we used Joy for the flow data feature extraction. Joy supports processing both packet capture files for offline processing and NetFlow flow records for on-line or live data processing. We implemented the machine learning algorithm and model using SciKit [11]. Scikit provides a number of tools that simplify the data analysis and include pre-built machine learning algorithms including Logistic Regression, Decision Trees, and Random Forest. The machine learning model labels the flows with the probability that the IP flow is video piracy. The labeled results are post-processed or filtered using a whitelist to remove any false-postives. The final result is stored in a database for report generation. The whitelists contains hosts that are well known sites that aren't sourcing video piracy (e.g. Netflix, Amazon, YouTube)

3.2. Case Study: Enterprise

An hour long packet capture file was captured from a small enterprise network that was known not to have any video piracy traffic. While performing the packet capture we introduced both pirated video and legitimate OTT video. The pirate video traffic included traffic from a IPTV set-top box connected to a pirate IPTV provider and multiple free IPTV sites that restream linear video. The legitimate video traffic included streamed video from NetFlix and ESPN.com. The capture file had the characteristics shown in Table 6.

Table 6 Enterprise Capture File Statistics

Capture File Size, bytes	16 GBytes
Number of Flows	13,503
Time Duration	69 minutes

We performed two tests. We ran the packet capture file through the model we generated with the Random Forest algorithm the first time using the full feature set (meta-data, byte distribution, packets lengths, and packet timing) and then a second time using just the meta-data feature set to compare the two results and to give us some kind of baseline for how the model should work with the sampled NetFlow in the second case study.

Table 7 Enterprise Traffic with Full Feature Set

	Labeled Piracy		Labeled Benign		Total
Unique end-points	249	3%	8453	97%	8,702
Number of flows	2,269	1%	345,596	99%	347,865
Number of bytes	7,353,501	0.47%	1,542,729,091	100%	1,550,082,592

Table 8 Enterprise Traffic with Meta-Data Only Feature Set

	Labeled Piracy		Labeled Benign		Total
Unique end-points	663	7%	8388	93%	9,501
Number of flows	13,503	4%	334,362	96%	347,865
Number of bytes	115,862,972	7.47%	1,434,219,620	93%	1,550,082,592

Consistent with our performance metrics, the Random Forest model generated using the full feature set peformed better than the Random Forest model generated with the smaller meta-data feature set.

Table 9 Top IP-Flows Identified with Random Forest and Full Feature Set

Hostname	Flow Bytes	Probability

Belgacom.be	6,499,379	0.95
Ip-streaming.net	219,701	0.94
Mivitec.net	135,214	0.93
Ucom.am	126,228	0.89
Worldstream.nl	117,343	0.98

We expected to find in the results worldstream.net as that was the host for the pirate IPTV service we used with the IPTV set-top box. In addition to worldsteam.nl, the model identified the hosts associated with other free pirate IPTV services – Belgacom.be, ucom.am, lofanga, and ip-streaming.net. We inspected the packet capture file and validated that IP flows associated with video and the hosts on the network that we used to view pirated video.

In addition, the model did NOT label the Netflix and ESPN.com video traffic as video piracy that we had running.

The model proved to be efficient at identifying pirate video flows that had the characteristics of the pirate IPTV service that we trained the model to look for. Later inspection of the Belgacom.be flows in the packet capture file, revealed that the IP address was associated with a residential ADSL modem. Further illustrating the performance of the Random Forest model with the full feature set.

As expected, the meta-data only feature set did not perform as well and falsely labeled a higher percentage of the traffic as pirated video. This was consistent with the lower accuracy, precision and higher false positive rate.

3.3. Case Study: Two Different Cable Operators

To further evaluate how well the machine learning model performed, we tested the model on flow data from two different cable operator residential broadband networks that provided a sampled NetFlow feed with. The flow data was formatted in NetFlow v9 [12] and v10 formats and included only the IP flow meta-data fields.

The byte count is the total number of bytes measured by the sampled NetFlow and therefore is lower than the actual number of bytes seen.

To reduce the false positives, we post-processed the results by running them through a whitelist filter as the labeled data included traffic that was either non-routable traffic (i.e. 0.0.0.0) or was from well-known sources such as ISPs, web hosts, and multi-tenant CDNs such as Amazon's CloudFront, Akamai, and Google and either labled them as "ignored" or "whitelisted". The post-processed results are shown in the tables below.

Table 10 Cable Operators Capture File Statistics

	Cable Operator #1	**Cabe Operator #2**

Capture File Size, Bytes	100 GBytes	3 GBytes
Time Duration	100 minutes	60 minutes

Table 11 Cable Operators Traffic with Meta-Data Only Feature Set

	Cable Operator #1				Cable Operator #2			
	Flow Pairs		**Bytes**		**Flow Pairs**		**Bytes**	
Labled Piracy	264,208	4.23%	15,065,361,900	0.58%	396,641	9.98%	1,707,959,736	1.56%
Labled Benign	152,087	3.26%	2,129,057,943,418	82.20%	123,402	3.11%	70,447,213,792	64.5%
Ignored	1,162,601	18.61%	336,119,008,254	12.98%	630,650	15.87%	31,149,334,147	28.52%
Whitelisted	4,616,809	73.90%	109,882,260,212	4.24%	2,823,460	71.05%	5,918,038,457	5.42%
Total	6.247,562	100%	5,345,901,876,081	100%	3,974,153	100%	109,222,545,132	100%

Subscribers	**Cable Operator #1**	**Cable Operator #2**
Pirating	19%	28%
Not Pirating	81%	72%
Total	100%	100%

The post-processed results found that a number of IP addresses on the ISPs networks had one or more flows that were labeled as “pirate”. The overall volume of traffic labeled “pirate” was low. It is important to remember that the machine learning classifier was trained to find only one form of streaming video, and therefore the numbers here do not reflect other potential forms of video piracy that may be occurring.

As we did with the enterprise data results, we looked at the end-points labled piracy and ranked the top hosts by volume and are shown in Table 12.

The results also included a number of false positives from online gaming sites, and streaming music. The machine learning classifier had not been trained to label these as benign, and this may have contributed to them being falsely labeled as a "pirate".

Table 12 Top Labled Pirate Hosts for Cable Operator #1 Case Study

Host
Tier 1 CDN
Online Gaming Platform #1
Online Gaming Platform #2
Hosting Provider #1
Hosting Provider #2
Hosting Provider #3
Hosting Provider #4

Table 13 Top Labled Pirate Hosts for Cable Operator #2 Case Study

Host
Online Gaming Platform #3
2nd Tier CDN
Streaming Music
Hosting Provider #6
Hoting Provider #7
On-line Gaming Platform #4
2nd Tier CDN

A number of the hosting providers and second tier CDNs labeled as "pirate" are consistent with other imperical analysises that has been performed where these same providers were seen to be hosting and serving pirated content.

Overall, the results were consistent with our machine learning performance metrics when using the Random Forest model and the meta-data only feature set in Table 5. The results were also consistent with the findings from the enterprise case study when using only the meta-data feature set. The results could be improved with by including additional benign traffic samples in the training data and with further post-processing by expanding the whitelisted hosts.

4. Applications

The supervised machine learning model can be applied to operators networks to classify traffic in a number of ways in the measurement and mitigation of video piracy [13] [14]. (Note, patent has been applied for some of the methods and processes as applied to video piracy described in this paper.) One application is the classification of traffic as part of a traffic measurement system. A second application is to use the results of the machine learning system to mitigate piracy traffic with network policy enforcement systems such as PacketCable Multimedia [15] or as input to the Policy Charge Rule Function (PCRF) in the Evolved Packet Core of 4G and 5G network architectures.

Figure 7 shows a system schematic for an implantation of a system using flow data as input to a machine learning system for identifying video piracy. The system utilizes multiple machine learning models or classifiers, both for identifying multiple forms of video piracy and for reducing false positives by identifying forms of legitimate OTT video traffic. The output of the system may then be fed to an operators network measurement and/or policy enforcement system.

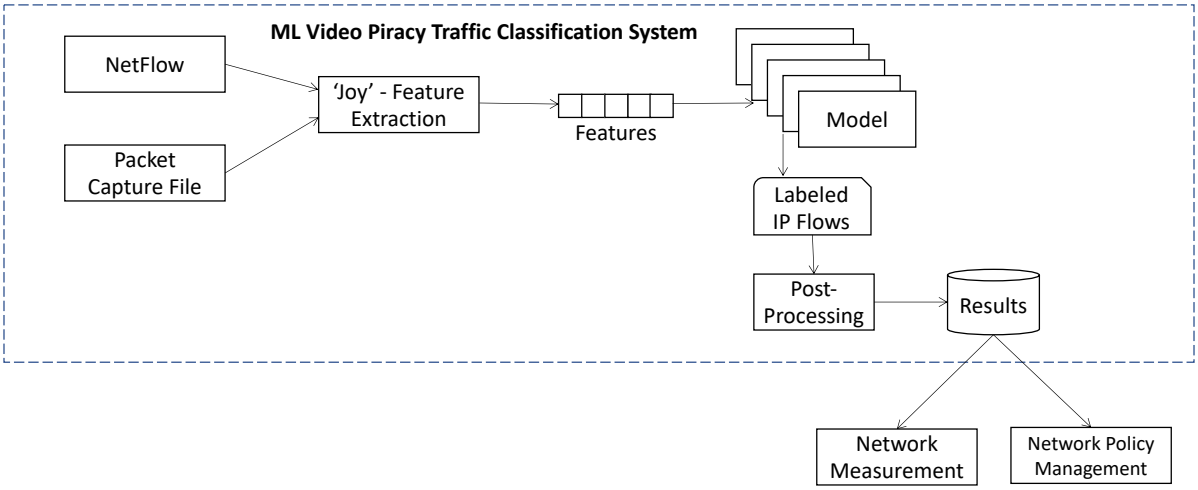

Figure 7 Video Piracy Traffic Classification System

Conclusion

In this paper, we showed that some forms of pirated video delivered as OTT IPTV can be efficiently identified using a machine learning based traffic classification system. Further we showed that the Random Forest model out performs other machine learning models such as Logistic Regression and Decision Trees when used in this fashion.

We also showed that an efficient machine learning model can be built to classify traffic using IP flow data that can be ingested directly from a packet capture file or from an IPFIX/NetFlow feed. And finally, we showed that using flow data that is enhanced with byte distribution, packet size, and packet timing information such as the proprietary fields included in NetFlow for Cisco's Encrypted Traffic Analytics can be used to build a machine learning model that has a high accuracy and a low false positive rate.

Abbreviations

ADSL	Asymmetric Digital Subscriber Line
AI	Artificial Intelligence
CDN	Content Delivery Network
DASH	Dynamic Adaptive Streaming over HTTP
DPI	Deep Packet Inspection
ESPN	Entertainment and Sports Programming Network
HBO	Home Box Office
HLS	HTTP Live Streaming
HTTP	Hyper Text Transfer Protocol
IPD	Initial Packet Data
IP	Internet Protocol
IPFIX	Internet Protocol Flow Information Export
IPTV	Internet Protocol Television
ISP	Internet Service Provider
MPEG	The Moving Picture Experts Group set standard for encoding and compressing video images
NBC	National Broadcasting Company
OTT	Over The Top
PCRF	Policy Charge Rules Function
SCTE	Society of Cable Telecommunications Engineers
SPLT	Sequence of Packet Lengths and Times
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
VOD	Video On Demand

Bibliography

[1] NCTA, "Broadband by the Numbers," [Online]. Available: https://www.ncta.com/broadband-by-the-numbers. [Accessed 19 July 2019].

[2] Parks Associates, "Parks Associates Forecasts $12.5B in Lost Revenue in 2024 Due to Pay-Tv and OTT Piracy and Account Sharing," [Online]. Available: https://www.parkassociates.com./blog/article/pr-07162019. [Accessed 19 July 2019].

[3] D. Jones and K. Foo, ""Analyzing the Modern OTT Piracy Video Ecosystem - NCTA Technical Papers" Tech Paper Database," 2018. [Online]. Available: http://www.nctatechnicalpapers.com/Paper/2018/2018-analyzing-the-modern-ott-piracy-video-ecosystm. [Accessed 19 July 2019].

[4] V. Mihajlovic, ""Global Internet Phenomena Spotlight: Video Piracy in North America." Sandvine," 13 December 2019. [Online]. Available: https://www.sandvine.com/blog/global-internet-phenomena-spotlight-video-piracy-in-north-america. [Accessed 19 July 2019].

[5] B. Anderson and D. Mcgrew, "Identifying Encrypted Malware Traffic with Contextual Flow Data," in *Proceedings of the 2016 ACM Workshop on Artificial Intelligience - AISec 16, 2016*, 2016.

[6] Cisco, "Cisco Encrypted Traffic Analytics," 2019. [Online]. Available: https://www.cisco.com/c/dam/en/us/solutions/collateral/enterprise-networks/enterprise-network-security/nb-09-encrytd-traf-anlytcs-wp-cte-en.pdf. [Accessed 19 July 2019].

[7] R. Pantos and W. May, "HTTP Live Streaming," August 2017. [Online]. Available: https://tools.ietf.org/html/rfc8216. [Accessed 19 July 2019].

[8] International Organization for Standardization, "SO/IEC 23009-1:2014 Preview Information technology -- Dynamic adaptive streaming over HTTP (DASH) -- Part 1: Media presentation description and segment formats," May 2014. [Online]. Available: https://www.iso.org/standard/65274.html. [Accessed 19 July 2019].

[9] T. T. Nguyen and G. Armitage, "A Survey of Techniques for Internet Traffic Classification using Machine Learning," *IEEE Communications Surveys & Tutorials,* vol. 10, no. 4, Fourth Quarter 2008.

[10] T. Hastie, R. Tibshirani and J. Friedman, "The elements of statistical learning: data mining, inference and prediction," Springer, 2009.

[11] Scikit-learn, "Scikit-learn Machine Learning in Python," [Online]. Available: https://scikit-learn.org. [Accessed 19 July 2019].

[12] Cisco, "NetFlow Version 9 Flow-Record Format," May 2011. [Online]. Available: https://www.cisco.com/en/US/technologies/tk648/tk362/technologies_white_paper09186a00800a3db9.html. [Accessed 19 July 2019].

[13] M. Tooley and W. Check, "Method and System for Detecting Pirated Video Network Traffic". United States of America Provisional Patent Applicaiton 62/740,569, 8 October 2018.

[14] M. Tooley and W. Check, "Method and System for Detecting Pirated Video Network Traffic". United States of America Patent Application 16/381,571, 11 April 2019.

[15] CableLabs, "PacketCable Specification - Multimedia Specification," 11 November 2015. [Online]. Available: https://specification-search.cablelabs.com/packetcable-multimedia-specification. [Accessed September 2019].

Endnotes

1 https://github.com/cisco/joy
2 Flusonic, TVHeadend, Xtremecodes
3 https://github.com/cisco/joy