Practical and Configurable Network Traffic Classification using Probabilistic Machine Learning

Jiahui Chen
University of Utah

UUCS-20-007

School of Computing
University of Utah
Salt Lake City, UT 84112 USA

30 April 2020

Abstract

Network traffic classification that is generally applicable and highly accurate is extremely valuable for many network security and management tasks. A flexible and easily configurable classification framework is ideal so it can be customized for use in many different networks. In this thesis we propose a highly configurable and flexible machine learning traffic classification method that relies only on statistics of sequences of packets to distinguish known or approved traffic from unknown traffic. Our method is based on likelihood estimation, provides a measure of certainty for classification decisions, and can classify traffic at adjustable certainty levels. Our classification method can also be applied in different classification scenarios, each prioritizing a different classification goal. We demonstrate how our classification scheme and all its configurations perform well on real-world traffic from a high performance computing network environment.

PRACTICAL AND CONFIGURABLE NETWORK TRAFFIC CLASSIFICATION USING PROBABILISTIC MACHINE LEARNING

by

Jiahui Chen

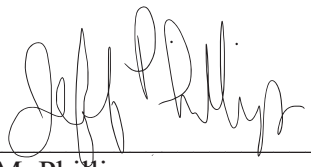
A Senior Honors Thesis Submitted to the Faculty of
The University of Utah
In Partial Fulfillment of the Requirements for the

Honors Degree in Bachelor of Science

In

Computer Science

Approved:



Jeff M. Phillips Thesis Faculty Supervisor	Ross Whitaker Director, School of Computing
Erin Parker Honors Faculty Advisor	Sylvia D. Torti, PhD Dean, Honors College

April 2020

Copyright © 2020
All Rights Reserved

ABSTRACT

Network traffic classification that is generally applicable and highly accurate is extremely valuable for many network security and management tasks. A flexible and easily configurable classification framework is ideal so it can be customized for use in many different networks. In this thesis we propose a highly configurable and flexible machine learning traffic classification method that relies only on statistics of sequences of packets to distinguish known or approved traffic from unknown traffic. Our method is based on likelihood estimation, provides a measure of certainty for classification decisions, and can classify traffic at adjustable certainty levels. Our classification method can also be applied in different classification scenarios, each prioritizing a different classification goal. We demonstrate how our classification scheme and all its configurations perform well on real-world traffic from a high performance computing network environment.

TABLE OF CONTENTS

ABSTRACT	ii
INTRODUCTION	1
BACKGROUND AND RELATED WORK	5
TRAFFIC REPRESENTATION METHODOLOGY	8
MACHINE LEARNING REPRESENTATION METHODOLOGY	11
RESULTS	21
CONCLUSION AND FUTURE WORK	32
REFERENCES	34

INTRODUCTION

The accurate and timely classification of network traffic is crucial to many network management and security tasks. Categorization of network traffic yields valuable information on a network's activity, and classification done in real-time enables this information to be quickly acted upon to ensure a secure and efficient network. Anomaly detection, quality of service monitoring, intrusion or attack detection, and resource allocation planning are all difficult network management tasks where traffic classification plays a critical role in solving. With the pervasive and diverse usage of the internet and online devices, large volumes of traffic from many different applications are constantly hosted on networks. Robust and flexible traffic classification is a difficult task due to the wide variety of traffic and dynamic nature of source applications. Traffic classification techniques have changed greatly over time, in reaction to changes in networking as a field.

Early and simple methods of traffic classification use port numbers to identify the traffic sources [1-3]. However, application port numbers became more unpredictable as more applications used obscure, protocol-based, or configurable ports, so port numbers were no longer a reliable source of classification [4-6]. In response to port-based classification becoming less effective, research turned to classification methods that use data packet inspection to find application or protocol signatures, i.e., patterns or data specific to the source application or protocol [4], [7-9]. These procedures require the ability to inspect packet payloads, so they are unable to classify encrypted traffic and also require high computational overhead and up-to-date application or protocol signatures to

match traffic with. These issues present considerable limitations to inspection-based classification.

Most current approaches to traffic classification use machine learning algorithms and statistical properties of traffic flows to categorize traffic. A flow is usually defined by all packets with the same 5-tuple: source/destination IP, source/destination port, protocol. The statistical properties of flows are referred to as features. Using statistical features of networking activity for classification avoids using ports or packet payloads, thereby remedying the limitations of port and payload based procedures. Machine learning techniques rely on the fact that different applications have differing networking behavior and patterns. These differences are represented in features then discovered and used to discern flows' classes by a machine learning model.

In this paper, we present a machine learning technique that uses statistics of subflows, or some series of packets from a flow, to classify traffic with a measure of certainty. We classify traffic using probabilistic learning with likelihood estimation and adjustable certainty levels, so we can classify traffic at higher or lower confidence levels based on network preferences. This approach allows network administrators to configure and use our classification so that it performs best on the most important traffic in their network. Our method can operate in three different classification scenarios: (1) classification performed with strict certainty thresholds resulting in known, unknown, and uncertain classification decisions; (2) classification with majority likelihood, eliminating any uncertain classification decisions; (3) incremental classification, where the classifier gathers information subflow by subflow, enabling the classifier to reach a classification decision as soon as possible. These different classification options along with adjustable

classification certainty levels make our technique very configurable, allowing it be be easily customized to best fit a network's needs.

We classify traffic into known and unknown classes where the known class consists of traffic from some group of applications approved for network usage, and the unknown class consists of traffic from any applications not in the known group. These class definitions fit well into real-world networks and take advantage of the fact that networks with specific intended application usage usually allow applications with similar functions and behaviors. The broad definition of the unknown class allows it to include a huge array of diverse application traffic, so the variation between unknown traffic and known traffic is bound to be greater than the variation within the known traffic class. The known class will generally contain specified applications with similar traffic and the unknown class will include a huge variety of applications that have different behaviors from the known traffic. Our method successfully finds and utilizes these differences for classification via machine learning. This class scheme is also flexible since the known class can be defined with any set of applications, allowing network administrators to define a custom known class for their network with applications that are allowed for usage on their network. Our technique is easily configured to fit a variety of network needs and is widely applicable to many real-world networks.

This work makes these main contributions:

- We present a probabilistic machine learning method that classifies traffic with a measure of certainty. We describe how the certainty of classification decisions can be easily configured to yield different results.

- We show that our method can be applied in 3 different classification scenarios, each prioritizing a different classification goal.
- We demonstrate how our method and all of its configurations can be used to effectively classify traffic in the Science DMZ [10] network setting.

BACKGROUND AND RELATED WORK

Traffic classification techniques using machine learning comprise two main components: the representation of network traffic and machine learning algorithm. From the vast existing research we present a brief overview of work relevant to ours.

Many different representations and statistical features of flows have been explored in previous work. Statistics on packet size, arrival times, and packet types have resulted in high classification accuracies when used with a wide variety of machine learning methods [5], [9], [11], [12]. These features can be calculated over all the packets in an entire flow or on some series of packets sampled from the flow [5], [11], [13], [14]. Research also exists on feature selection techniques used to reduce the number of features needed for classification and to find optimal features that result in the best statistical representation of network traffic [12], [15]. In these works, packet size statistics and discrete feature values were found to enable classification accuracies of 93% and above for multiple machine learning algorithms [12].

Calculating features over an entire flow is not ideal for timely classification, prompting more practical network traffic classification methods that classify sequences of packets in a flow. Using features on only the first few packets of flows was found to yield reasonable classification results [11-12]. Earlier work also found that using a sequence of packets, or subflows, of as few as 25 packets can result in classification precision and recall of above 95% [13]. This subflow work was expanded upon by [14], finding that classification performance is not affected by the position of the subflow within the overall flow or the direction of the packets. Other work has explored

different methods of selecting subflows that are especially representative of the statistics of the overall flow for training, so that training requires minimal processing of only the optimal examples [14], [16]. In [13], [14], [17] the length of the subflow (value of N) results in a trade-off between classification performance and processing requirements, with higher values of N leading to better classification but requiring more processing time and memory [13], [14], [16].

Many different machine learning algorithms have been used for traffic classification. Early work used traditional supervised learning methods, that classify traffic into pre-defined classes, include decision trees and Bayesian analysis techniques [5], [13], [18], [19]. These methods have been shown to perform classification at accuracies above 95% on various sets of applications [5], [18]. Unsupervised and semi-supervised learning methods, where traffic is grouped based on similarity rather than explicitly classified into a class, have also been explored in [6], [20-23]. Clustering unlabelled or partially labelled traffic resulted in classification accuracies of 90-93% [6], [20]. Recent methods use deep learning, with supervised classification performed by convolutional neural networks and recurrent neural networks, and unsupervised learning of traffic representations and traffic imitation performed by auto-encoders and generative adversarial networks [11]. Various architectures of neural nets used for classification have achieved high accuracies of up to 96% [17].

Most of this existing work classifies traffic by mapping it to an application, application type, or protocol. A few classify traffic into known and unknown classes by discerning a specific, known application or group of applications from other traffic which may include many other applications [13-14], [24]. Our work uses this latter scheme of

known and unknown classification as it is less explored, more flexible, and widely applicable. For example: known traffic could be defined as a broad set of non-malicious activities for a well-protected, low-risk network but a small set of specifically approved applications for a network with less security designed for specialized uses only. The flexibility of this known vs. unknown classification brings additional challenge, as our classification method must be robust enough to perform well on many different sets of known applications.

In addition to addressing the more challenging task of classifying traffic into flexible known and unknown classes, we consider classification in the Science DMZ network setting which has not been previously explored. A Science DMZ is a subnetwork, usually part of a university network, that is configured and designed to optimize the usage of high-performance scientific computing applications [10]. This network definition fits well with our known vs. unknown classification, as a Science DMZ is intended to host traffic from specific scientific computing applications and no other traffic. Our traffic dataset is sourced from University X's Science DMZ, which allows us to evaluate our method on realistic high-performance computing traffic. Our approach performs classification at or near 100% accuracy on representative Science DMZ traffic, and we additionally evaluate on a more challenging traffic dataset with reasonable results to show that our method is generalizable.

TRAFFIC REPRESENTATION METHODOLOGY

A feature vector representation of network traffic is necessary in order to use machine learning for classification. We present the format and statistical features used to represent traffic captures as a dataset for machine learning.

A. Use of Subflows

Practical traffic classification needs to occur quickly, so that a network's allowed traffic is not delayed by classification and unknown or unapproved traffic can be effectively stopped. Since entire flows can last long periods of time and require high amounts of memory to process fully, it is ideal for a classification method to only use some portion of packets from a flow. Using subflows, defined as some N packets taken from any point in a flow, for classifier training and evaluation was first introduced in [13]. We use N-packet subflows to represent our traffic, where N=25, 100, 1000 due to these aforementioned advantages. These values of N were discussed, experimented upon extensively, and found to be sufficient subflow lengths in [13-14], [16], with the larger values of N leading to better classification performance but requiring more processing time and memory. Our statistical features are calculated over each N-packet subflow and all of our flows are split into N-packet subflows for classification.

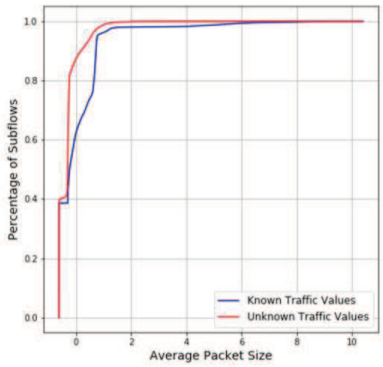
An additional advantage that using subflows gives our classification approach is the ability to gather multiple data points per flow. Each subflow gives our classifier some statistical data on the overall flow, so it can use each subflow to increase or decrease certainty in a classification decision for the overall flow. Thus, our classification

approach can gain valuable classification progress for each encountered subflow, effectively utilizing the breakdown of flows into subflows.

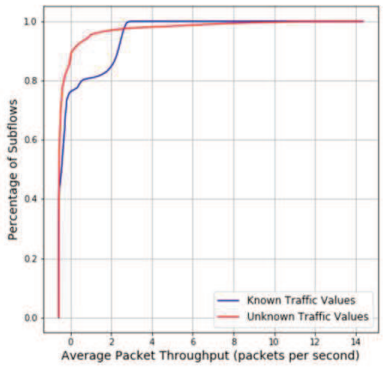
B. Statistical Features of Traffic

We calculate the 14 statistical features below over all packets in each subflow: Total Bytes, Largest Packet Size, Smallest Packet Size, Number of TCP ACKs, Minimum Advertised Receive Window, Maximum Advertised Receive Window, Standard Deviation of Packet Size, Average Packet Size, Average Packet Inter-Arrival Time, Standard Deviation of Packet Inter-arrival Time, Maximum Packet Inter-arrival Time, Minimum Packet Inter-arrival Time, Average Packet Throughput (packets per second), Average Byte Throughput (bytes per second). Each subflow is represented by a 14-element data point where each element is a feature value, and is subsequently processed by our machine learning method as a 14-dimensional vector.

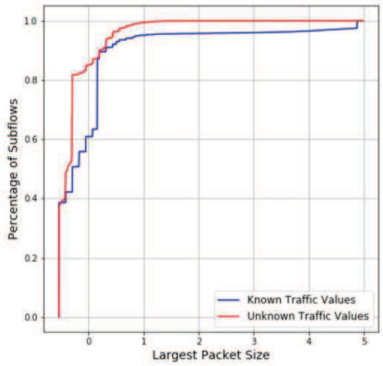
These features are sourced from a broader set of flow statistics used in previous work that were found to achieve the best network traffic classification performance [6], [9], [12], [18], [24]. For our feature selection process, we graphed the cumulative density function (CDF) of feature value distributions for our known and unknown traffic datasets to ensure that the features we use capture notable differences between known and unknown traffic. Fig. 1 shows example CDFs for various feature values.



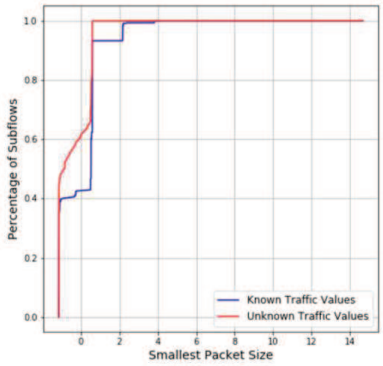
(a) Avg. Packet Size



(b) Avg. Packet Throughput



(c) Largest Packet Size



(d) Smallest Packet Size

Fig. 1: Feature Value CDFs for 100-Packet Subflows

MACHINE LEARNING METHODOLOGY

In this section we discuss the reasoning and data analysis leading to the formulation of our machine learning approach. We also describe the classification method and the various ways it may be applied. Fig. 2 shows our methodology's components, pipeline, and various usage options.

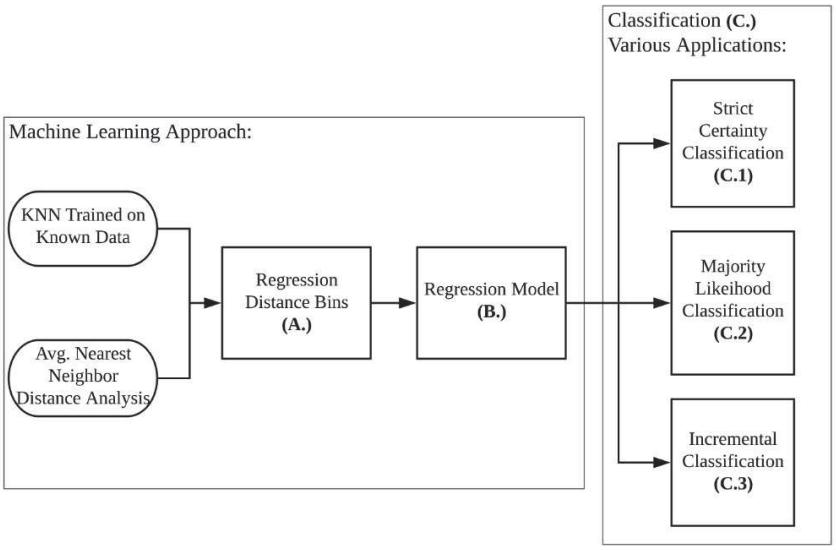


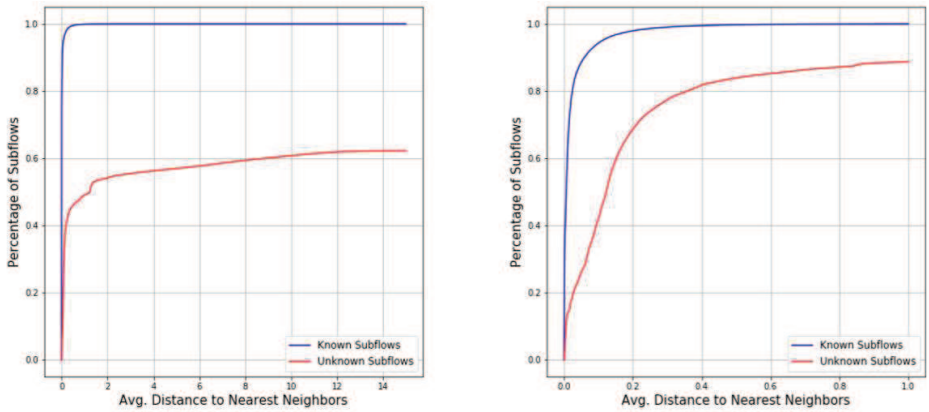
Fig. 1: Machine Learning Approach and Applications (with corresponding thesis sections)

A. Average Nearest Neighbor Distances Analysis

Our classification method builds a regression model of the average distance of a traffic data point to its K nearest known traffic data points. Using the average distance to K nearest known traffic data points provides an intuitive measure of distance to the known data based on k-Nearest-Neighbor (KNN) classification, where a points gets a label based on the label of its k nearest neighbors [25]. We use Euclidean distance and K=3 for all our experiments. This measure of distance allows our method to model the

similarity or dissimilarity of the statistics of sub-flows in the known and unknown classes and perform classification based on distances between values.

An intuitive assumption that known traffic will behave more similarly to other known traffic than unknown traffic is the underlying reasoning to our classification approach and the usage of distances. This approach is reasonable, since the known class is defined as some set of applications that are approved for network usage, and often networks with specific intended usage will allow applications with similar functions and behaviors. Additionally, the unknown class is defined as any traffic not belonging to the approved applications, so it can encompass a huge array of diverse traffic which is bound to have more variation in behavior than between known applications.



(a) 25-Packet Subflows (b) 100-Packet Subflows

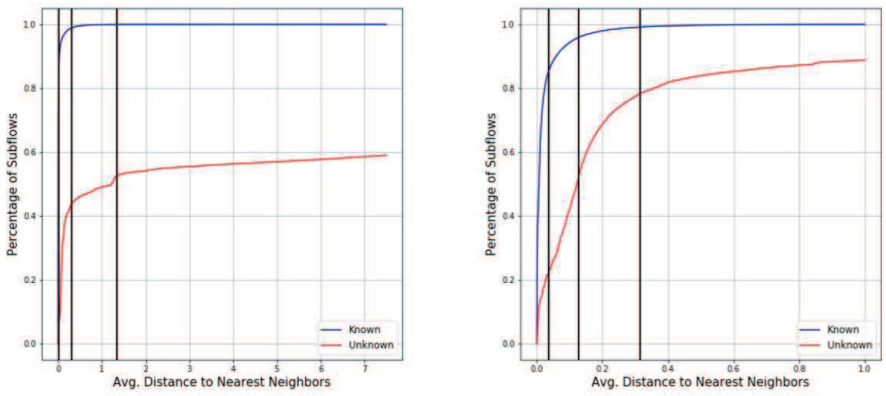
Fig. 3: CDFs of Average Nearest Neighbor Distances

Fig. 3 shows cumulative density functions (CDFs) of average nearest neighbor distances of known and unknown traffic to a KNN training set of only known traffic, for 25 and 100 packet subflows. From these CDFs, it can be seen that the distance distributions are notably different between known and unknown traffic. Generally a much

higher percentage of known traffic distances are near 0, which aligns with our assumption that known traffic will behave similarly to other known traffic. The unknown traffic distances have a much larger range with smaller percentages of their distances at small values; this also supports the assumption that unknown traffic has more varied behavior that's more dissimilar to known traffic. Thus, we believe this distance-based technique is very powerful and flexible since it can adapt to different sets of known traffic. Some unknown traffic is still similar to known traffic, as seen by the percentages of unknown subflows with very small distances to the known subflows present in the CDFs. So our classification task is still quite challenging.

The CDFs also show the distance value ranges that the highest frequencies of known and unknown sub-flows fall into. X-axis ranges where a CDF slope is steep indicate high counts of sub-flows that have those distances. Regions where the CDFs of known and unknown traffic distances have the most differing slope are of special interest, as they show distance ranges that are common in one class's traffic but not in the other. We propose to utilize these distance ranges where known and unknown traffic distributions behave differently to perform our classification. The overall idea is to create distance bins, or ranges, that include distances where there are considerably different counts of known and unknown subflows, then to assign class likelihoods to these bins so that a subflow is assigned a likelihood based on its average distance to its K nearest known traffic points. This method is described in more detail in the next subsection.

Through this straightforward, density-based analysis of distance distributions of known and unknown traffic, we can discover distance ranges where there exists the most difference between classes. Some example distance bins are visualized with vertical lines marking their boundaries in Fig. 4. This analysis is valuable as it yields a quantifiable measure of difference between known and unknown traffic.



(a) 25-Packet Subflows (b) 100-Packet Subflows

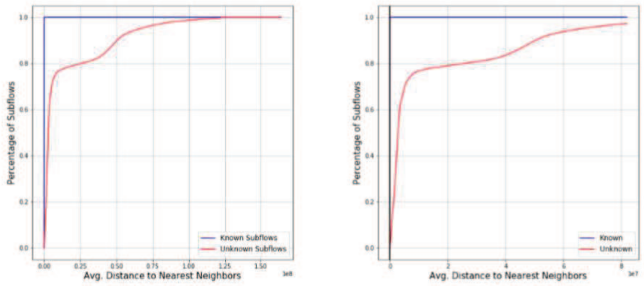
Fig. 4: Vertical lines mark distance ranges where known and unknown traffic distributions differ the most

B. Regression Model of Average Nearest Neighbor Distance

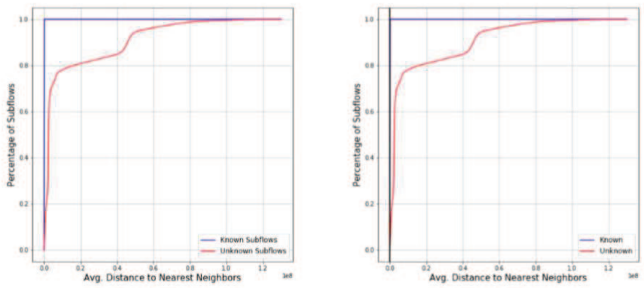
After discovering distance bins where there exists considerable difference in the amount of known and unknown subflows with distances in the bin's range, we use these bins to form a regression model.

We train the regression model by assigning known and unknown class likelihoods to each bin. These class likelihoods can be thought of as estimated probabilities that a subflow with an average nearest neighbor distance in the bin's range is from a flow of either class. For class likelihood, we simply use the count of subflows in the regression

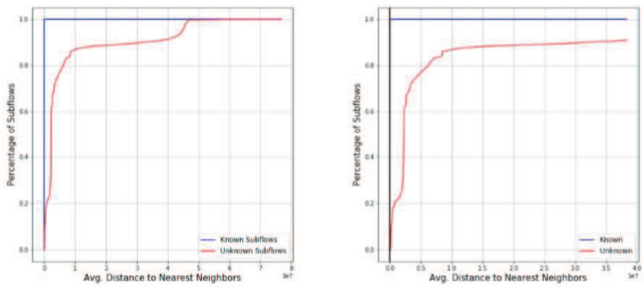
training set that have distances in the bin and are of the corresponding class, divided by the total number of subflows in the regression training set that have distances in the bin. The class label counts of subflows in the regression training set are the same, to ensure fair class likelihoods. Thus, high frequencies of a certain class in a bin translates into that bin having a high likelihood for that class. Figures 5 and 6 show all of our experimental datasets' CDFs with regression distance bins. Note that for the Science DMZ Unknown Data, the known and unknown distances ranges are drastically different so only 2 bins are defined with the distance boundary being the maximum known subflow distance.



(a) 25-Packet Subflows

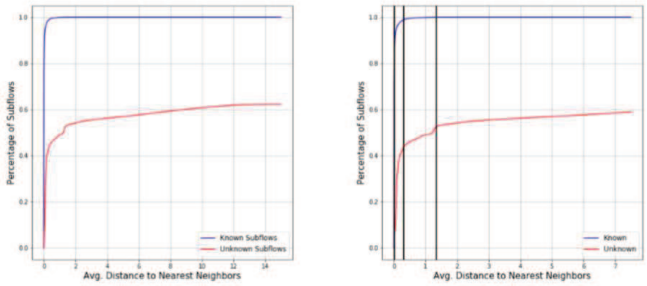


(b) 100-Packet Subflows

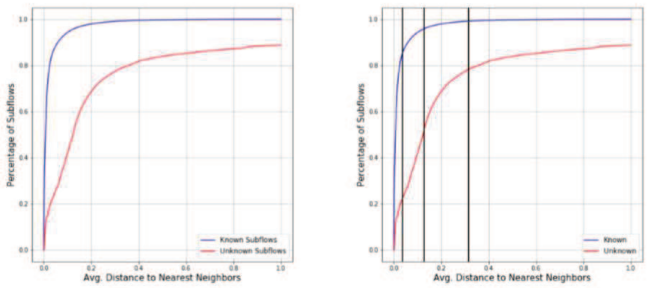


(c) 1000-Packet Subflows

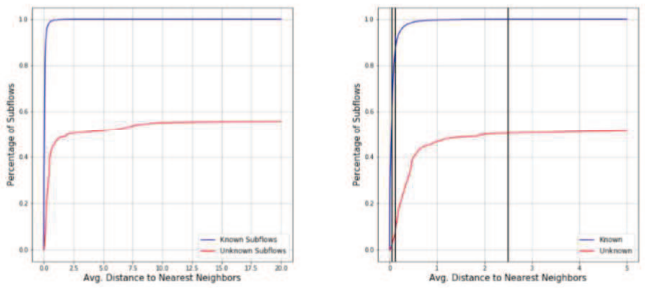
Fig. 5: **Science DMZ Dataset**: CDFs with Regression Distance Bins Marked as Vertical Lines



(a) 25-Packet Subflows



(b) 100-Packet Subflows



(c) 1000-Packet Subflows

Fig. 6: **General Dataset**: CDFs with Regression Distance Bins Marked as Vertical Lines

C. Classification Via Likelihood Estimation and Certainty Threshold

For classification using our regression model, the class likelihoods of individual subflows, assigned by their distance bins, are used to create joint class likelihoods over a sequence of subflows $s_1, s_2, \ldots$ all belonging to the same flow. The ratio of known and unknown joint likelihoods are then compared to a configurable, user-given certainty threshold to classify the flow from which the subflows come from.

To create the class joint likelihoods over multiple subflows, we assume independence and take the product of all subflow likelihoods of the same class. The class joint likelihoods can be used as estimated probabilities that the sequence of subflows is of the corresponding class. The flow likelihoods can also be used to form a likelihood ratio, which we use as a measure of certainty for classification. The likelihood ratio is a fraction of the class likelihoods, indicating how much larger one class likelihood is than the other. For example, if the known class likelihood is 0.95 and the unknown class likelihood is 0.05 then the likelihood ratio is $\frac{0.95}{0.05}$. This indicates that under our model, we are 95% certain that the flow is known, as the marginal probability that the flow is known, given all the subflows the classifier has seen, is 0.95. However, likelihoods of the numerator and denominator may not sum to 1, and in general the joint ones will not. But if the ratio is still 19, i.e., $\frac{0.019}{0.001}$, then the confidence is still 95%.

In more detail, we divide the distances into t bins $\mathrm{B}_1, \mathrm{B}_2, \mathrm{B}_3, \ldots \mathrm{B}_\mathrm{t}$ (we either use $t = 2$ or $t = 4$ in experiments); each one is fit with an empirical probability $\mathrm{p_K(i)} = \frac{|\mathrm{X_K} \cap \mathrm{B_i}|}{|\mathrm{X} \cap \mathrm{B_i}|}$ that a subflow is known, given that it falls in bin B_i. This is the fraction of known training data $\mathrm{X_K}$ that is in bin B_i compared to all training data $X \cap \mathrm{B_i}$ in that bin. The empirical probability $\mathrm{p_U(i)}$ of being unknown is computed symmetrically with X_U in

place of X_K. Given a subflow $\mathrm{s_j}$ let $B(\mathrm{s_j})$ map to the index of the bin it is in, e.g., $B(\mathrm{s_3}) = 2$ if s_3 is in bin B_2. We estimate the likelihood a series of observed subflows $\mathrm{s_1}, \mathrm{s_2}, \ldots, \mathrm{s_m}$ are known as $\widehat{L_K} = \mathrm{p_K}(\mathrm{B}(\mathrm{s_1})) \cdot \mathrm{p_K}(\mathrm{B}(\mathrm{s_2})) \cdot \ldots \cdot \mathrm{p_K}(\mathrm{B}(\mathrm{s_m}))$, and similarly for unknown $\widehat{L_U}$, and their ratio as:

$$\frac{\widehat{L_K}}{\widehat{L_U}} = \frac{p_K(B(s_1)) \cdot p_K(B(s_2)) \cdot \ldots \cdot p_K(B(s_m))}{p_U(B(s_1)) \cdot p_U(B(s_2)) \cdot \ldots \cdot p_U(B(s_m))}$$

By using a certainty threshold for classification, we can easily enforce the likelihood required for a flow to be classified. The use of different certainty thresholds for each class is also possible, which may be useful if the certainty of classification should be different between known and unknown traffic. For example, if a network is using our classification to block unknown traffic and wants to avoid disrupting allowed traffic, our technique would be applied with a very high certainty threshold for unknown classifications to ensure blocked traffic is classified as unknown with high confidence. The ease of adjusting classification certainty allows the certainty to be used as a parameter for classification. Different certainties can yield different classification accuracies depending on the underlying known and unknown traffic, and certainty can be a cross-validated hyperparameter that optimizes classification performance.

This likelihood estimation classification method can be applied in 3 different scenarios that we describe below and evaluate in our experiments:

1) Strict Certainty Classification:

In this classification scenario, flows are classified as known, unknown or uncertain. If the know or unknown likelihood ratio reaches the desired certainty level, then the flow is classified as known or unknown. However, it is possible that neither likelihood ratio

reaches the certainty level, so the flow is considered uncertain as its subflows do not yield a likelihood of high enough certainty for either class. Uncertain flows are indicative of traffic that is not similar enough to either class for a confident classification.

This designation of uncertain flows may be useful as a means of filtering and monitoring traffic, enabling uncertain flows to be found and tracked. Uncertain flows may be used for further analysis with a more specific method of classification or inspected as the potential source of network issues. The amount of traffic classified as uncertain is configurable with the certainty level, as higher certainties result in more uncertain decisions.

2) Majority Likelihood Classification

For this classification scenario, if neither of the class likelihood ratios have reached the certainty level after all available subflows are seen, then the flow is classified as the class with the larger likelihood. This scenario results in no uncertain flow classifications since all uncertain flows are classified by their majority likelihood. This approach allows some flows to be classified with less certainty than the given certainty level, but generally increases accuracy in our experiments and is a viable option if uncertain flows are not desired.

3) Incremental Classification

In this classification scenario, the class likelihood ratios are updated with each encountered subflow's likelihoods, and classification occurs immediately once either class likelihood ratio reaches the given certainty level. Incremental classification takes full advantage of our usage of subflows, utilizing each sequence of packets in a flow to gain information on the flow and classify it after seeing the least amount of subflows

possible. A classification decision is made as soon as possible, so this scenario prioritizes classification speed. In our Results section, we show that this scenario results in very fast classification after encountering a small fraction of subflows with excellent unknown detection capabilities. Note that incremental classification can use strict certainty or majority likelihood classification when making its classification decisions.

RESULTS

A. Dataset

To demonstrate and evaluate our classification method, we use the Science DMZ subnetwork setting. A Science DMZ is a subnetwork, usually part of a university campus network, that is configured and designed to optimize the usage of high-performance scientific computing applications [10]. These subnetworks are commonly used by researchers to transfer large datasets and have performance optimizing security measures or other policy differences to enable faster data transfers. This setting fits well with our known vs. unknown classification, as a Science DMZ is intended to host traffic of specific scientific computing applications and no other traffic. Fig. 7 shows the location of our traffic capture tap in a university's Science DMZ and Table 1 shows size statistics of our dataset.

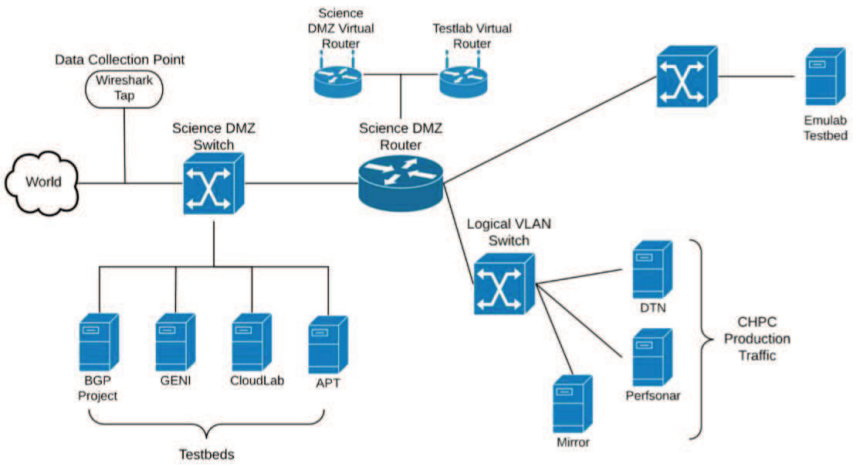


Fig. 7: Data Collection Point in the Science DMZ Sub-network

TABLE I: Dataset Statistics

	Globus	FDT	rclone	Mirror	WIDE
Bytes (GB)	51.6	129	82.1	42.6	30.48
Flows	185	72	12,292	2,239	1,112,554

Note that we have different numbers of known and unknown flows, so our experimental accuracies are calculated separately for each label. All of our traffic is TCP and uses IPv4. We randomly select 80% of our data for training and the rest for evaluation and ensured that the flows in the train and evaluation sets are mutually exclusive. The specifics and application breakdowns of our known and unknown datasets are below:

Known Dataset:

Our known traffic is from 3 widely used large file transfer applications: Globus [26-27], FDT [28], and rclone [29]. We consulted domain experts and system administrators at the Center for High Performance Computing at the University of Utah to ensure that these 3 applications are commonly used by science researchers on the Science DMZ. The Globus captures were of ongoing file transfers between Globus endpoints at a university and various other universities in the United States. The FDT traffic was generated by moving DNA sequencing datasets from the Hunstman Cancer Institute to and from Data Transfer Nodes [30] in the Science DMZ. The rclone traffic was generated by transferring ESnet test datasets [31] to and from Google Drive. We verified with domain experts that our usage of FDT and rclone to generate traffic was consistent with their common usage in science research workflows, to ensure that our data is representative of real FDT and rclone traffic.

Unknown Dataset:

For our unknown traffic, we use the Mirror and WIDE datasets. The Mirror dataset consists of random captures from a mirror server on the University of Utah's Science DMZ subnetwork that hosts repositories and other downloadable content. The WIDE

dataset consists of captures, performed on the same dates as the Mirror captures, from the WIDE Traffic Archive [32]. The WIDE captures are from the main internet exchange link and internet service provider transit link of the WIDE organization [32].

For all of our classification experiments, we train and evaluate our models using 2 different datasets. The known dataset always consists of the Globus, FDT, and rclone datasets but we use 2 different unknown class definitions: Science DMZ and General. The Science DMZ unknown class consists of only the Mirror traffic dataset, which was captured from the University of Utah's Science DMZ subnetwork but does not contain known application traffic. This approach allows us to simulate traffic classification in a realistic Science DMZ setting.The General unknown class consists of both the Mirror and WIDE datasets, resulting in a much broader, more diverse unknown traffic class since WIDE's traffic is not from the same network and contains many more flows. Using this more varied unknown traffic allows us to evaluate how well our classification method generalizes when classifying more challenging, varied traffic.

B. Strict Certainty Classification Results

To evaluate Strict Certainty classification, we perform our likelihood estimation classification and require a flow's class likelihood ratio to reach the given certainty threshold to be classified as known or unknown. Flows with class likelihood ratios that do not surpass the certainty threshold are considered uncertain. In our experiments, we perform classification using 25%, 50%, 75%, and 100% of subflows in each of the test set flows in order to evaluate classification performance when varying amounts of packets in flows are seen. We require at least 15 subflows in a flow portion to perform

classification. We also perform classification on features calculated over subflows of different packet lengths, using 25, 100, and 1000 packet subflows. We use these different combinations of percentage-defined subflow subsets and differing lengths of subflows to thoroughly evaluate classification in many situations where different portions of flows are seen.

Science DMZ Dataset:

Table II shows classification accuracies on the Science DMZ dataset, when using a strict certainty threshold of 95%. Our accuracies are extremely high across all subflow sizes and subflow percentage subsets, with a minimum accuracy of 94.28% and most experiments reaching 100% accuracy. These results show that the unknown traffic is very different from the known application traffic and our method can successfully find and utilize these differences for classification. Additionally these results show that our classification performs better on smaller subflow sizes, which is ideal because classification can be done using less packets from a flow. No flows were classified as uncertain across all experiments.

TABLE II: Science DMZ Dataset: Strict Certainty and Majority Likelihood Accuracies

Percentages of Subflows	**25%**	**50%**	**75%**	**100%**
Known Accuracies:				
25-Packet Subflows	100	100	100	100
100-Packet Subflows	100	100	100	94.28
1000-Packet Subflows	96.97	96.97	96.97	96.97
Unknown Accuracies:				
25-Packet Subflows	100	100	100	100
100-Packet Subflows	100	100	100	100
1000-Packet Subflows	100	100	100	100

General Dataset:

Fig. 8 shows classification accuracies on the General dataset, using the same strict certainty threshold of 95%. These accuracies are generally lower than the Science DMZ accuracies, which is expected since the General dataset contains unknown traffic that is more varied and similar to the known traffic, resulting in a more challenging classification task. The unknown accuracies are considerably higher than the known accuracies, with the experimental combination imposing the smallest amount of packets seen (25% of flows with 25-packet subflows) resulting in the best classification performance of 94.87%. This result shows that our classification method is ideal for detecting unknown traffic and can do so after seeing a small percentage of packets in flows. Table III shows the percentages of unknown flows that were considered uncertain at the 95% certainty threshold. Percentages of uncertain unknown flows

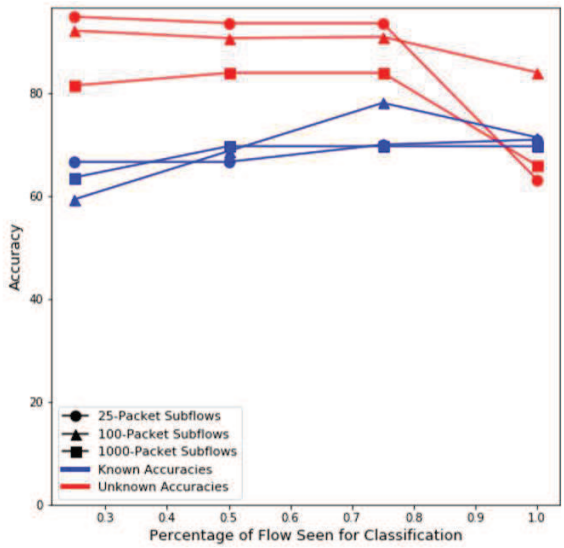


Fig. 8: General Dataset: Strict Certainty Classification Accuracies

TABLE III: Unknown Flow Uncertainty Percentages

Percentages of Subflows:	**25%**	**50%**	**75%**	**100%**
25-Packet Subflows	0.74	1.16	0.497	35.62
100-Packet Subflows	2.81	2.81	1.33	13.49
1000-Packet Subflows	3.7	2.47	2.47	30.17

are low overall, with a maximum of 35.62%, and the amount of uncertain flows grows as higher percentages of subflows are used for classification. Only one experiment resulted in any uncertain known flow classifications: 3% of known flows with 1000-packet subflows were classified as uncertain.

On both datasets, a small number of flows were considered uncertain even when a small percentage of subflows are seen. This shows that even if a high certainty for classification is enforced and not all packets in a flow are seen, our method can classify a large majority of flows. Our results indicate that smaller numbers of packets seen actually increases the amount of certain classifications that can be made, as seeing all available subflows of flows resulted in the highest uncertain flow percentages out of all 3 subflow lengths.

C. Majority Likelihood Classification Results

To evaluate Majority Likelihood classification, we perform our likelihood estimation classification to classify a flow as known or unknown if that flow's corresponding class likelihood ratio reaches the given certainty threshold. If after all available subflows are seen and the flow has no class likelihood ratio that has reached the certainty threshold, then the flow is classified as whichever class has the larger, or majority, likelihood estimate. We use the same percentage-defined subflow subsets and differing lengths of subflows as the Strict Certainty Classification experiments (25%, 50%, 75%, and 100% of a flow's subflows each with 25, 100, and 1000 packet subflows).

Science DMZ Dataset:

There were no flows classified as uncertain in this dataset using 95% certainty, so all flows reached 95% certainty for either class across all amounts of subflows seen for classification. This means that there are no differences in accuracy between Strict Certainty and Majority Likelihood classification for all experiments on the Science DMZ dataset, and Table II shows the unknown and known flow classification accuracies for Majority Likelihood classification.

General Dataset:

Fig. 9 shows classification accuracies on the General dataset when using a certainty threshold of 95%. The known flow accuracies do not notably differ from the Strict Certainty classification known accuracies because there were very few known flows classified as uncertain, so classifying uncertain flows by their majority likelihood mostly impacted unknown flow accuracies.

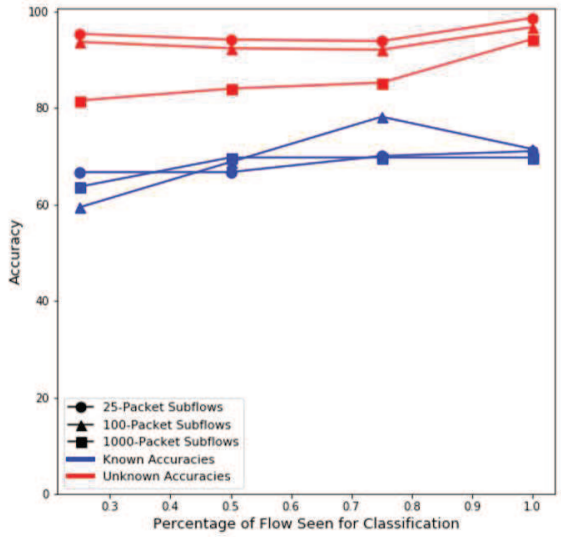


Fig. 9: General Dataset: Majority Likelihood Classification Accuracies

The unknown accuracies when full flows are classified using Majority Likelihood are notably higher than those from using Strict Certainty classification, with full flow classification with 25-packet subflows reaching 98.6% accuracy. These increases in accuracy correspond to the percentages of flows that were considered uncertain when

using Strict Certainty classification. This indicates that classification by the larger class likelihood is an effective way to classify traffic that is not similar enough to either class for a classification at the certainty required by the given threshold. The unknown accuracies when percentages of flows were seen did not drop and most slightly increased in comparison to accuracies from Strict Certainty classification. These results indicate that classifying traffic using majority likelihood is a viable and simple option, allowing improved classification accuracy and the elimination of uncertain flows.

D. Incremental Classification Results

To evaluate Incremental classification, we update a flow's class likelihoods and check if the given certainty threshold is reached for every encountered subflow. Classification of the flow occurs immediately once a class likelihood reaches the certainty threshold, and subflows are encountered in chronological order of packet arrival; so flows are classified as soon as possible.

Thus, these experiments allow us to evaluate how well our classification performs when reaching a classification decision in the fastest manner possible. We use both Strict Certainty and Majority Likelihood classification with this Incremental classification scheme, where Strict Certainty will allow for uncertain flows and Majority Likelihood will classify all flows as known or unknown even if no certain decision is reached after all subflows are seen. We evaluate on all lengths of 25, 100, and 1000 packet subflows.

Science DMZ Dataset:

Fig. 10 shows classification accuracies on the Science DMZ dataset when using Incremental classification and a 95% certainty threshold.

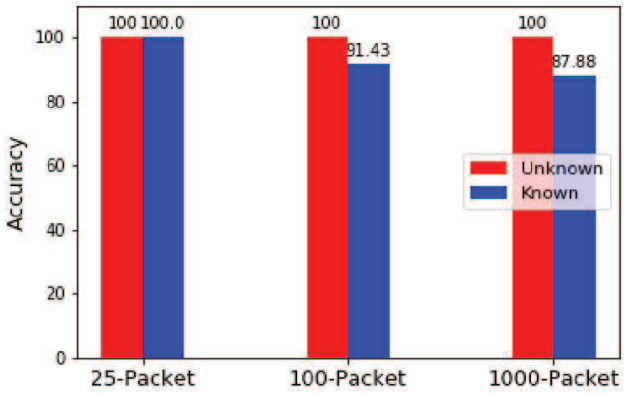
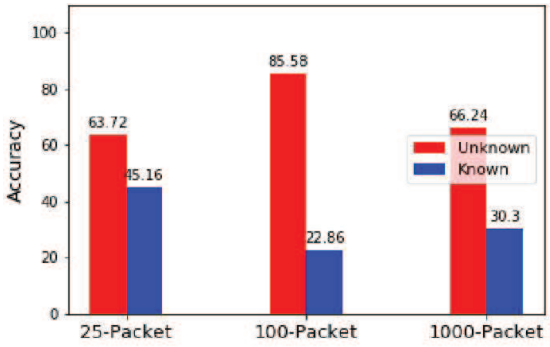


Fig. 10: Science DMZ Dataset: Incremental Classification

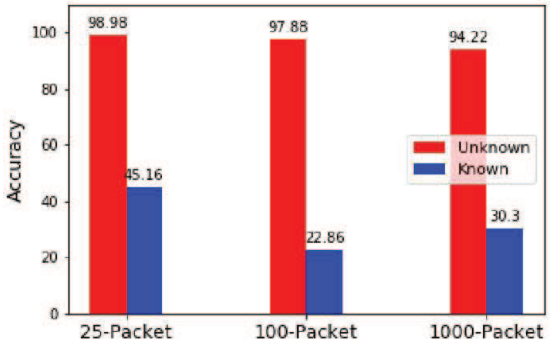
Just as in non-incremental classification, there were no uncertain flows so the accuracies from using Strict Certainty and Majority Likelihood classification are the same. The accuracies are very high, with all unknown classification accuracies being 100% and classification on 25-packet subflows reaching 100% for both known and unknown accuracies. Larger subflow lengths result in drops in known classification performance but performance is still good, with a minimum accuracy of 87.88% using the largest subflow length of 1000. Higher accuracies on smaller subflows is most ideal, since this shows that our method performs the best when using the fewest amounts of packets. These results show that our method still performs very well even when making a classification decision after seeing the fewest amount of subflows possible.

General Dataset:

Fig. 11 shows classification accuracies from Incremental classification with a 95% certainty threshold on the General Dataset. When Incremental classification is used with Strict Certainty classification, accuracies for known and unknown flows drop considerably when compared to accuracies from non-incremental Strict Certainty classification. For Incremental classification used with Majority Likelihood classification, the known accuracies are lower but the unknown accuracies are slightly higher than the results from non-incremental Majority Likelihood classification. No known flows during Incremental classification with Strict Certainty were classified as uncertain, as switching to Majority Likelihood did not change the known accuracies. However, a considerable portion of unknown flows were classified as uncertain during Incremental classification with Strict Certainty, and they were able to be successfully classified as unknown when using Majority Likelihood. These results support the viability of Majority Likelihood classification as a method to improve classification performance.



(a) Strict Certainty Classification



(b) Majority Certainty Classification

Fig. 11: General Dataset: Incremental Classification

On the General dataset, the mean percentage of subflows seen before making a flow classification across all subflow lengths was 3.45% for known flows and 54.3% for unknown flows. So the known decisions were made very quickly after seeing a small number of subflows. The known decisions were also all certain, indicating that the certainty threshold was reached in a small amount of subflows. This result shows that the known flows have subflows with extreme likelihoods that quickly form a highly confident flow likelihood estimate. Unfortunately, these likelihoods result in mostly wrong classifications, which indicates that most of the known subflows had higher unknown likelihoods. The unknown decisions were made on average after a bit over half of a flow's subflows were seen, which is fairly timely, but the resulting drop in accuracy when compared to non-incremental Strict Certainty classification on 50% of subflows indicates that incremental classification is not ideal for unknown traffic. From results across both datasets, it seems that Incremental classification generalizes more weakly than non-incremental classification.

CONCLUSION AND FUTURE WORK

In this thesis, we introduced a machine learning method for traffic classification that uses statistics on sequences of packets, called subflows, to classify traffic as known or unknown with a measure of certainty. Our technique creates a regression model of traffic distances to known traffic and uses this model for probabilistic classification using likelihood estimation. This method of classification allows traffic to be classified at an easily configurable certainty level and in three different ways. If used with Strict Certainty thresholds, flows are only classified as known or unknown if they can be classified at the given certainty level, and our method can find uncertain flows that are not similar enough to either class. If used with Majority Likelihood, all flows are classified as known or unknown by allowing some flows to be classified with whichever class likelihood estimate is higher rather than strictly requiring the certainty level. If used in an Incremental classification manner, each subflow seen updates the flow's likelihood estimate and classification of a flow occurs after seeing the fewest number of subflows possible.

We evaluated our technique on traffic from the Science DMZ subnetwork domain [10], as it naturally fits our class scheme and has not been used as a traffic classification setting before. We also evaluate on a more general, challenging dataset to ensure that our method can generalize well. Our results showed that our classification performs very well in the Science DMZ setting, able to reach 100% accuracy for all classification options. On the general dataset, we maintained high accuracy on unknown traffic classification, reaching up to 98.98%, though known classification accuracies dropped. Our method was

shown to perform well even when only seeing partial flows, reaching accuracies up to 100 on the Science DMZ dataset and 95.3 on the general dataset when only a fourth of all subflows in a flow are used for classification. With Strict Certainty classification, very few flows are considered uncertain even when requiring 95% certainty and seeing partial flows. The use of Majority Likelihood classification was shown to correctly classify flows deemed uncertain in Strict Certainty classification, improving classification performance. The Incremental classification approach reached classification decisions very quickly after seeing small amounts of subflows and maintained high accuracies on the Science DMZ dataset.

Out of all the classification scenarios, Incremental classification accuracies dropped the most between the Science DMZ and General dataset results, so further work could be done to achieve more generalizable Incremental classification. Known accuracies are also generally lower than unknown accuracies, because distance bins in higher distance ranges have very large unknown likelihoods. If a known flow has subflows that fall into these higher distance bins, it is difficult for the flow's likelihood ratio to favor the known class, especially if the high-distance subflows are consecutive and classification occurs after seeing a small amount of subflows. Further work on forming the regression model and bin likelihoods could improve classification performance by regularizing bin likelihoods so that their label likelihood values are more similar. The usage of likelihood priors for distance bins could help standardize subflow likelihoods, avoiding the extreme likelihoods that caused known accuracies to drop.

REFERENCES

Name of Candidate:	Jiahui Chen
Birth date:	December 1, 1998
Birth place:	Guangzhou, China
Address:	4828 Bull Run Drive Plano TX, 75093

REFERENCES

[1] S. Saroiu, K. P. Gummadi, R. J. Dunn, S. D. Gribble, and H. M. Levy, "An analysis of internet content delivery systems," *SIGOPS Oper. Syst. Rev.*, vol. 36, no. SI, p. 315–327, Dec. 2003. [Online]. Available: https://doi.org/10.1145/844128.844158

[2] S. Sen and Jia Wang, "Analyzing peer-to-peer traffic across large networks," *IEEE/ACM Transactions on Networking*, vol. 12, no. 2, pp. 219–232, 2004.

[3] D. Moore, K. Keys, R. Koga, E. Lagache, and K. C. Claffy, "The coral-reef software suite as a tool for system and network administrators," in *Proceedings of the 15th USENIX Conference on System Administration*, ser. LISA '01. USA: USENIX Association, 2001, p. 133–144.

[4] A. W. Moore and K. Papagiannaki, "Toward the accurate identification of network applications," in *Passive and Active Network Measurement*, C. Dovrolis, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 41–54.

[5] T. 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, pp. 56–76, Fourth 2008.

[6] J. Erman, M. Arlitt, and A. Mahanti, "Traffic classification using clustering algorithms," in *Proceedings of the 2006 SIGCOMM Workshop on Mining Network Data*, ser. MineNet '06. New York, NY, USA: ACM, 2006, pp. 281–286. [Online]. Available: http://doi.acm.org/10.1145/1162678.1162679

[7] P. Haffner, S. Sen, O. Spatscheck, and D. Wang, "Acas: Automated construction of application signatures," in *Proceedings of the 2005 ACM SIGCOMM Workshop on Mining Network Data*, ser. MineNet '05. New York, NY, USA: Association for Computing Machinery, 2005, p. 197–202. [Online]. Available: https://doi.org/10.1145/1080173.1080183

[8] S. Sen, O. Spatscheck, and D. Wang, "Accurate, scalable in-network identification of p2p traffic using application signatures," in *Proceedings of the 13th International Conference on World Wide Web*, ser. WWW '04. New York, NY, USA: Association for Computing Machinery, 2004, p. 512–521. [Online]. Available: https://doi.org/10.1145/988672.988742

[9] T. Karagiannis, K. Papagiannaki, and M. Faloutsos, "Blinc: Multilevel traffic classification in the dark," in *Proceedings of the 2005 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, ser. SIGCOMM '05. New York, NY, USA: Association for Computing Machinery, 2005, p. 229–240. [Online]. Available: https://doi.org/10.1145/1080091.1080119

[10] "Science-DMZ." [Online]. Available: http://fasterdata.es.net/science-dmz/

[11] S. Rezaei and X. Liu, "Deep learning for encrypted traffic classification: An overview," *CoRR*, vol. abs/1810.07906, 2018. [Online]. Available: http://arxiv.org/abs/1810.07906

[12] H. Kim, K. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, and K. Lee, "Internet traffic classification demystified: Myths, caveats, and the best practices," in *Proceedings of the 2008 ACM CoNEXT Conference*, ser. CoNEXT '08. New York, NY, USA: ACM, 2008, pp. 11:1–11:12. [Online]. Available: http://doi.acm.org/10.1145/1544012.1544023

[13] T. T. t. Nguyen and G. Armitage, "Training on multiple sub-flows to optimise the use of machine learning classifiers in real-world ip networks," in *Proceedings. 2006 31st IEEE Conference on Local Computer Networks*, Nov 2006, pp. 369–376.

[14] T. T. T. Nguyen, G. J. Armitage, P. Branch, and S. Zander, "Timely and continuous machine-learning-based classification for interactive ip traffic," *IEEE/ACM Transactions on Networking*, vol. 20, pp. 1880–1894, 2012.

[15] H. Zhang, G. Lu, M. T. Qassrawi, Y. Zhang, and X. Yu, "Feature selection for optimizing traffic classification," *Comput. Commun.*, vol. 35, no. 12, p. 1457–1471, Jul. 2012. [Online]. Available: https://doi.org/10.1016/j.comcom.2012.04.012

[16] T. T. T. Nguyen and G. Armitage, "Clustering to assist supervised machine learning for real-time ip traffic classification," in *2008 IEEE International Conference on Communications*, 2008, pp. 5857–5862.

[17] M. Lopez-Martin, B. Carro, A. Sanchez-Esguevillas, and J. Lloret, "Network traffic classifier with convolutional and recurrent neural networks for internet of things," *IEEE Access*, vol. 5, pp. 18 042–18 050, 2017.

[18] A. W. Moore and D. Zuev, "Internet traffic classification using bayesian analysis techniques," in *Proceedings of the 2005 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, ser. SIGMETRICS '05. New York, NY, USA: Association for Computing Machinery, 2005, p. 50–60. [Online]. Available: https://doi.org/10.1145/1064212.1064220

[19] M. Shafiq, X. Yu, A. A. Laghari, L. Yao, N. K. Karn, and F. Abdessamia, "Network traffic classification techniques and comparative analysis using machine learning algorithms," in *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, 2016, pp. 2451–2455.

[20] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, "Semi-supervised network traffic classification," *SIGMETRICS Perform. Eval. Rev.*, vol. 35, no. 1, p. 369–370, Jun. 2007. [Online]. Available: https://doi.org/10.1145/1269899.1254934

[21] P. Casas, J. Mazel, and P. Owezarski, "Minetrac: Mining flows for unsupervised analysis semi-supervised classification," in *2011 23rd International Teletraffic Congress (ITC)*, 2011, pp. 87–94.

[22] J. Zhang, X. Chen, Y. Xiang, W. Zhou, and J. Wu, "Robust network traffic classification," *IEEE/ACM Transactions on Networking*, vol. 23, no. 4, pp. 1257–1270, 2015.

[23] J. Erman, A. Mahanti, M. Arlitt, I. Cohen, and C. Williamson, "Offline/realtime traffic classification using semi-supervised learning," *Performance Evaluation*, vol. 64, pp. 1194–1213, 10 2007.

[24] R. Baker, R. Quinn, J. Phillips, and J. Van der Merwe, "Toward classifying unknown application traffic," in *Proceedings. DYnamic and Novel Advances in Machine Learning and Intelligent Cyber Security DYNAMICS'18*, 2018.

[25] P. Cunningham and S. J. Delany, "k-nearest neighbour classifiers," University College Dublin, Tech. Rep. Technical Report UCD-CSI-2007-4, March 2007.

[26] I. Foster, "Globus online: Accelerating and democratizing science through cloud-based services," *IEEE Internet Computing*, vol. 15, no. 3, pp. 70–73, 2011.

[27] B. Allen, J. Bresnahan, L. Childers, I. Foster, G. Kandaswamy, R. Kettimuthu, J. Kordas, M. Link, S. Martin, K. Pickett, and S. Tuecke, "Software as a service for data scientists," *Commun. ACM*, vol. 55, no. 2, p. 81–88, Feb. 2012. [Online]. Available: https://doi.org/10.1145/2076450.2076468

[28] "The Fast Data Transfer Tool: Overcoming Limitations to High Performance Transfers Over the Wide Area Network," Tutorial, 2017. [Online]. Available: https://indico.hep.caltech.edu/event/174/

[29] "rclone." [Online]. Available: http://https://rclone.org/

[30] "Science DMZ: Data Transfer Nodes." [Online]. Available: https://https://fasterdata.es.net/science-dmz/DTN/

[31] "ESnet Data Transfer Nodes." [Online]. Available: https://fasterdata.es.net/performance-testing/DTNs/

[32] K. Cho, K. Mitsuya, and A. Kato, "Traffic data repository at the wide project," in *USENIX 2000 FREENIX Track*. USENIX, 2000.

Name of Candidate:	Jiahui Chen
Birth date:	December 1, 1998
Birth place:	Guangzhou, China
Address:	4828 Bull Run Drive Plano TX, 75093