INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 9, Issue 2 - V9I2-1172)
Available online at: https://www.ijariit.com

Machine Learning Based Network Intrusion Detection for Cyber Security

Mounika Maity
mounikamaity8989@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

B. Pramod
pramodbobburi@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

N. Masthan Valli
nadendlamasthan123@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

Y. Pranathi
pranathiy1970@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

M. Mallikarjuna
varunteja390@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

V. Sathyendra Kumar
tpo.aits.vsk@gmail.com
Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh

ABSTRACT

Machine Learning-based systems act on flow features derived through exporting flow procedures. The notable emergence of Machine Learning and Deep Learning (DL) based reports presuppose that the flow of information, such as the average packet capacity, is gleaned from every packet. On common devices, However, when packet sampling is unavoidable, flow exporters are frequently used in practice. Since the flow of information is derived from a sampled group of the packets rather than the entire traffic stream, the usefulness of Machine Learning-based results with the use and existence of such samplings is still up for debate. In this study, we are going to investigate in what ways the effectiveness and performance of these ML-based are affected by packet sampling. Our suggested evaluation method is resistant to various flow export stage settings, in contrast to earlier studies. Hence, it can provide a robust evaluation even in the presence of sampling.

Keywords: *Supervised Learning, Anomaly Detection, Intrusion Detection, Random Forest, Neural Network, Decision Tree, Support Vector Machine, ML techniques, e-learning, and Principal Component Analysis.*

1. INTRODUCTION

In the twenty-first century, packet switched networks have quickly given way to IP-based networks as the development of telecommunications networks has progressed. It is now feasible to connect IP-based voice and data across apps and services thanks to this advancement [1]. Expansion of communication networks has increased technology viability, but it has also created unwelcome new possibilities. Threats that formerly only affected fixed networks can now affect wireless access networks.

Due to the reality that threats are developing and becoming more sophisticated, more intelligent security systems are

required. Firewalls and malware scanners, two common security tools, are at capacity. Applications for network monitoring such as analyzing the flow of data, performance monitoring and intrusion detection have been emerged. Due to the constant growth in network traffic volume and speed, these technologies are becoming more and more common[2]-[5].

Due to the efficiency of flow-based network monitoring over full-packet monitoring methods like deep packet inspection (DPI), Instead than looking at individual packets, efficiency is obtained by looking at the flow records. Only 0.2% of packet exporting technologies (like NetFlow) cause a network load as a result of flow record collecting and export. Flow records are used as inputs by flow based Network Intrusion Detection Systems (NIDSs), which assess whether a certain flow is malicious or not. Recent studies have recommended. In this paper, a method for creating effective IDS that makes use of the random forest classification algorithm and principal component analysis (PCA) is proposed. Whereas the random forest will aid in classifying while the PCA will assist in organizing the dataset by lowering its dimensionality.

For flow-based NIDSs, there is a substantial body of all the Machine Learning(ML) and Deep Learning(DL) outcomes. In terms of their high detection rates, several systems have shown promise (DRs). To the best of our knowledge, the bulk of cutting-edge solutions use the assumption that flow monitoring and its records are calculated from the flow and traffic, but generally they are gathered and calculated from the collected sample of packets [6]-[9].

The effectiveness of cutting-edge in ML or else DL based methods used in real-world applications are therefore uncertain. The suggested machine aims to eliminate the ongoing problems caused by the earlier efforts. The proposed machine includes the two methods that are crucial for aspect evaluation, and the random woodland is the alternative. The random woodland set of rules, which provide both the detection price and the fake alarm price in a more sophisticated manner than SVM.

II. EXISTING SYSTEM

Due to a lack of knowledge about data visualization, it is a bit difficult to deploy machine learning algorithms in the current system. In the current approach, constructing models is done by mathematical computations, which can be very difficult and time-consuming. We employ machine learning tools from the Scikit-Learn toolkit to get around all of this. Machine learning is becoming more and more prevalent, and there are now machine learning and traditional computer techniques. The Performance Analysis of Intrusion Detection Systems prediction works that are linked to this part are discussed, along with why machine learning techniques are superior to older ones. Model development is carried out using the current technique, which has a certain flow. The systems now in use are SVM and ANN algorithms. The result, however, is inaccurate and requires a huge memory.

1. Support Vector Machine (SVM)

Support Vector Machine (SVM) is one of the linear or supervised machine learning method that is mostly used for classification and regression applications. Additionally, this approach is more effective at addressing a wider range of real-world issues.

A hyperplane is typically created and a line is referred using a straightforward technique to categorize data. This approach is more adapted to the numerous issues that have arisen in real-world applications, such as helping to classify complex and simple language, classify and segment images, recognize handwriting characters, and a large portion of biological and auxiliary sciences.

Apart from this, the SVM algorithm seeks to find the best category to quickly categorize fresh data points in the future by detecting a path as well as decision boundaries. In this way classes can be created in each and every n-dimensional space. The main intention of support vector machine technique is to identify a hyper plane that clearly categorizes all the data points in an N-dimensional space of conceivable hyper planes.

Possible Hyper Planes:

There are numerous different Hyper planes that can be selected to differentiate the two kinds of data points. The biggest range, or the greatest distances involved among the data points through both the classes, is what we are looking for in a plane. If only two varieties of input characteristics are involved then the hyper plane is merely a line. If there are three input

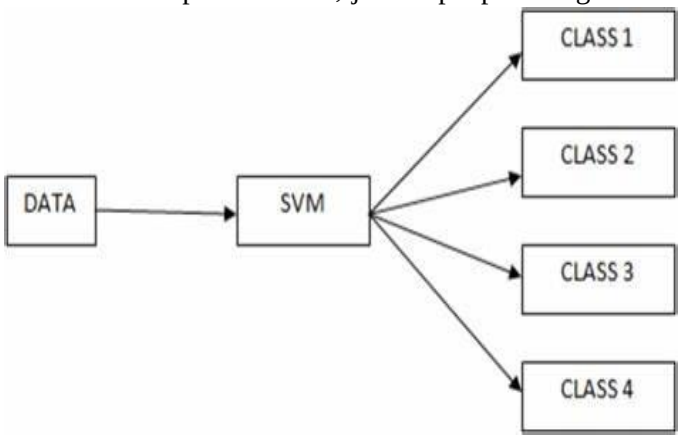
characteristics, the hyper plane collapses into a two-dimensional plane.

Figure 1: ***Support Vector Machine***

2.Artifiicial Neural Network (ANN)

Artificial neural network (ANN) is one of the well known component of a computing systems which is made to replicate the information evaluation and processing in the human brain. ANNs are capable enough for learning self and they can help in providing the results which are better as well as more information can be made available. An ANN contains large number of elements to be processed. Artificial neurons are one among them and these are connected with each other at the nodes. Inputs and outputs are used for processing the data.

Backpropagation, which is also known as Backward Propagation of the inaccuracies, is nothing but learning a set of rules which ANN uses to filter the output solutions, just as people use guidelines and various procedures to generate



relevant results or outputs. In order to recognize a novel pattern, ANNs display a complex relationship between inputs and outputs. These synthetic neural networks are employed in a variety of tasks, such as voice, image, and machine translation recognition and medical diagnosis. By utilizing these kinds of technologies, one can define the distribution's solutions in a way that is both practical and cheap. An alternative to using the entire dataset is to use the samples of data to produce results using the ANN. One can enhance present data analysis techniques because ANNs have highly developed prediction abilities.

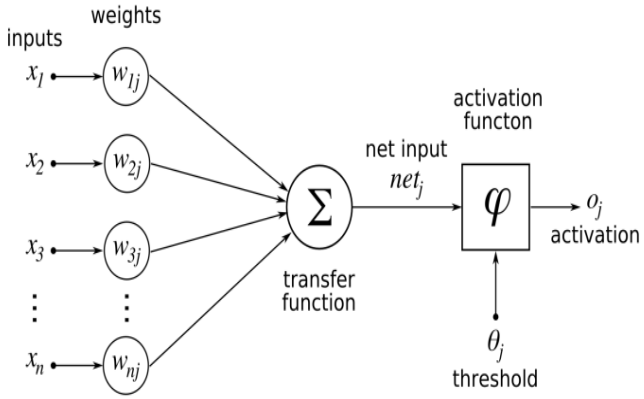


Figure 2: ***Artificial Neural Network***

III. LITERATURE SURVEY: Related work

In order to recognize intrusions, the Intrusion Detection System (IDS) is used to find system attacks and for the identification of attackers. There are several online security dangers as a result of the development of wireless communication. Previously, a variety of Machine Learning techniques were applied to IDS in an effort to improve the generated results for the purpose

of detecting the intrusions and improving the accuracy of IDS [10].
The Principle Component Analysis (PCA) and Random Forest classification algorithm have been suggested as a method for creating effective IDS in this study. The PCA will aid in data organisation by lowering the dataset's dimensionality, while the random forest will aid in categorization. From all these results, it can be finalized and concluded that the method which is suggested performs more accurately and efficiently than other methods like SVM, Naive Bayes, and Decision Tree [11]-[15]. Our research resides at the nexus of two subdomains that are strongly related to one another:

(1) investigations made in NIDS when there is existence of sampling of all the relevant data; and (2) using various approaches of NIDS related to Machine learning based techniques.

Unbalanced datasets can be handled successfully using the Random Forest classifier. The basic researches undergoing with ML-based NIDS that took into account of real-world scenarios where sampling is necessary is built on the framework for evaluation we have proposed [16]. The suggested method for ML model evaluation illustrates the performance improvements brought about by resolving data imbalance and highlights the significance of choosing the appropriate grouping measurement for multi-class classifications to prevent results that are favorably skewed for the purpose of achieving real world data constancy that is unaffected by sampling. The impact of these samplings on NIDS is then examined using the suggested evaluation methodology. Here such ML classifiers are assessed on sample flow of data rather than Flow records acquired from entire traces [10].

Flow records are used as inputs by Flow based Network Intrusion Detection Systems (NIDSs), which then analyze each flow to see if it is malicious or not. The percentage of accurately predicted flows over the entire number of flows is the detection rate for the particular harmful category.
With macro averaging, the overall detection rate is a metric that is combined across many harmful categories. The fraction of benign (i.e., normal) traffic that is mistaken for hostile categories is known as the false alert rate. In such cases of their high detection rate, these solutions have shown encouraging outcomes (DRs). To the finest of our knowledge, the bulk of cutting-edge solutions use the assumption that Flow records are calculated from complete traffic, but in reality they are gathered from various samples of data packets[11].

Multiple measures like accuracies, performance, true positive result rate, negative result rate and detection of false rates of alarm can be used to assess multiclass attack categorization. Because of this, we turn to the most basic yet useful data, such as high range of detection and fake alarm rates detection. The percentage of accurately anticipated flows over all flows, as measured for a certain harmful category, is the detection rate. A macro averaging technique is used to aggregate the overall detection rate across many harmful categories. False/Fake alert rate is nothing but proportion of innocent (i.e., regular) traffic which is mistaken for malicious category.

Recent studies that have demonstrated the effectiveness of certain combination concepts, such as boosting (or arcing), bagging, random subspaces, or more recently, random forests, have piqued interest. In order to maximise the ensemble's generalisation performance, the effectiveness of merging classifiers depends on the capacity to consider the complementarity between individual classifiers.

IV. PROPOSED METHODOLOGY

Many machine learning models have been put forth to categorize attacks as being legitimate or not. Additionally, comparable research that have developed models for evaluation frequently ignore the heterogeneity and magnitude of the data. We therefore suggest machine learning-based strategy that combines the DECISION TREE and RANDOM FOREST, with a novel method of pre-processing the information for features modification. Various ML based algorithms like Random Forest, SVM, ANN, Decision Tree classifiers are used to anticipate network infiltration. Among all of them RANDOM FOREST model provides the best accuracy method for removing bias and deviations from stability while carrying out classifier tests.

Random Forest Classifier:

Deterministic tree makes up a random forest algorithm. Through bootstrap aggregation, the algorithm random forest tests and trains the "forest" it generates. Bagging, an ensemble meta-algorithm, improves the accuracy of machine learning systems. The algorithm chooses the outcome based on the predictions made by the decision trees.

To create predictions, the output from multiple trees is aggregated or averaged. [17]-[19]. The outcome is more precisely predicted when there are more trees.
The Random Forest Algorithm has the following advantages over the Decision Tree Algorithm:
• It is useful and more accurate.
• It effectively handles missing data.
• It addresses the problem of over fitting in decision trees and can generate an acceptable prediction without hyper-parameter modification.

Figure 3 shows how decision trees are used in random forest algorithm.

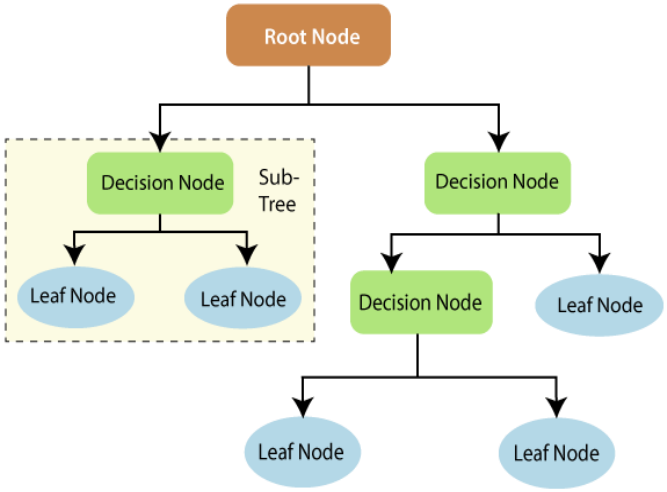


Figure 3*: Applying Decision trees in Random Forest*

The main variation between these two algorithms, decision tree method and random forest algorithm is it randomly chooses the nodes at the root and groups all those nodes. In this way random forest uses bagging approach for forecasting. The number of trees must be predetermined (ii). There is no assurance that all the trees in the ensemble will work together to form a committee with this type of method, where each tree is introduced to the ensemble independently. This demonstrates that the "classical" RF induction procedure, which adds randomised trees at random to the ensemble, is not the most effective method for creating precise RF classifiers.

A) Level Of Performance For ML-Based NIDS

Various measures are used to assess the selection within the framework of Network Intrusion Detection are first explained in the sections that follow. Afterward, we describe and defend our NIDS evaluation setup for testing.

ANALYSIS OF SAMPLERS:

In two ways, sampling has an impact on NIDS: (1) it reduces the visibility of the flow, and (2) it reduces the accuracy of the estimated flow records.

a: Visibility of Flow

To determine the proportion of flows that are still visible after sampling, we create a flow visibility metric. No flow record is exported if no packets from the flow is sampled. As a result, the NIDS does not assess flow [20]. A binary metric, flow visibility can either be noticed or not. When atleast one packet is sampled, visibility is 100%, ensuring that a Flow record is created for the current data flow. When there is no such packet gathered from the flow, flow data will not export and the value will be 0. A binary metric, flow visibility can either be noticed or not. When at least one packet is sampled, visibility is 100%, ensuring that the Flow record is created for the current Flow.

b: Reliability of Flow

The reliability of a record is reduced as a result of data loss when predicting the flow data from sampled packets. Consequently, ML classifier's performance when studying to differentiate between various attack or good traffic categories. Therefore, higher quality flow records from a particular sampler will result in improved performance of NIDS [21]. In the other words, if the identical classifier of NIDS was constructed within two distinct flow data that correlate with two distinct

sampling procedures. Thus, sampler produces the superior performance of the NIDS which means that it is more appropriate for the NIDS purposes than the counterpart of it. As a result, we infer Flow record evaluation of quality from the performance of this NIDS rather than using an exclusive metric.

B) Key Metrics for Classifier Evaluation

Multiple measures like accuracies, performance, true positive result rate, negative result rate and detection of false rates of alarm can be used to assess multiclass attack categorization [22]. From record level measurements, our evaluation generates flow level metrics. Therefore, we turn to the most basic yet useful measurements such as high range of detection and fake alarm rates detection. And the percentage of flows which are predicted accurately among all the flows is the detection rate for the given harmful category. With macro averaging, the complete detection of the accuracy is one of the parameters which is combined among many harmful categories. The fraction of benign traffic that is mistaken for hostile categories is known as the false alert rate. With the existence of sampling, only a portion of the Flow will be regarded. calculating the high range of detection and fake alarm rates detection. Consideration for such alert rates are the number of Flows which are detected for each as a ground truth category.

Decision Tree Algorithm:

An internal node represents a feature (or attribute), a branch represents a decision rule, and each leaf node represents the result in a decision tree, which resembles a flowchart. The root node in a decision tree is the first component from the top. It gains the ability to divide data according to characteristic values. Recursive segmentation is the process of repeatedly dividing a tree. This framework, which resembles a flowchart, aids in decision-making. It is a flowchart-like visualization that perfectly replicates how people reason. Decision trees are simple to comprehend and interpret because of this.

Any decision tree algorithm's fundamental principle is as follows:

1. To divide the records, choose the best trait using trait Selection Measures (ASM).
2. Break the dataset up into smaller groups and make the characteristic a decision node.
3. Recursively repeats this procedure for every child to begin building the tree as long as one of the requirements is met:

• The same property value applies to each and every tuple.
• There are several no more characteristics left.
• No more occurrences exist.

V. CONCLUSION:

For correctly assessing ML-based algorithms, we proposed a methodology. According to our most reliable memories, we are recognized for being the first to propose the Flow Level NIDS evaluation model, which can be applied even when sampling is taking place. We demonstrated how correcting the training-data imbalance led to a startling performance improvement using the proposed evaluation method. The results of sampling experiments show that 50% of harmful flows are not exported, even at a moderate 1/10 sample rate. In general, such sampling will reduce the performance of NIDS, according to our study of the viability of machine learning-based techniques in the presence of a sample. In addition, we discovered that this sampling technique can work better when sources like the flow cache of the measuring device are accessible. Future research should also look for the effects Sampling on NIDS which is based on anomalies. Another drawback of current research and study is that the size of the sample is relatively very small. The number of cache sizes we could test in our flow cache tests was similarly restricted to four. As a result, larger research with various dimensions may be conducted in the future. Investigating the impact of sampling using high sampling rates is one approach. The inclusion of a thorough list of sample strategies and ML/DL approaches is another direction. Additionally, doing these tests on numerous, diverse datasets results in more convincing justifications for the impact of Sampling at the throughput.

VI. REFERENCES:

[1] R. Hofstede et al., “Flow Monitoring Explained: From Packet Capture to Data Analysis With NetFlow and IPFIX,” IEEE Commun. Surv. Tut., vol. 16, no. 4, pp. 2037–2064, 2014, doi: 10.1109/COMST.2014.2321898. 20 VOLUME 4,

2016 This work is licensed under a Creative Commons Attribution 4.0 License. For more information,see https://creativecommons.org/licenses/by/4.0/ This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2021.3137318, IEEE Access Author et al.: Preparation of Papers for IEEE TRANSACTIONS and JOURNALS

[2] A. Sperotto and A. Pras, "Flow-based intrusion detection," 12th IFIP/IEEE Int. Symp. on Integr. Netw. Manage. (IM 2011) and Workshops, 2011, pp. 958–963, doi: 10.1109/INM.2011.5990529.

[3] B. Claise, B. Trammell, and P. Aitken, "Specification of the ip flow information export (ipfix) protocol for the exchange of flow information," Internet Requests for Comments, RFC Editor, RFC 7011, Sep. 2013. [Online]. Available: http://www.rfc-editor.org/rfc/rfc7011.txt

[4] M. F. Umer, M. Sher and Y. Bi, "Flow-based intrusion detection: Techniques and challenges", Comput. & Secur., vol. 70, pp. 238–254, 2017, doi: 10.1016/j.cose.2017.05.009.

[5] B. Claise, "Cisco systems netflow services export version 9," Internet Requests for Comments, RFC Editor, RFC 3954, Oct. 2004. [Online]. Available: http://www.rfc-editor.org/rfc/rfc3954.txt

[6] V. Kumar. "Network Intrusion Detection on UNSWNB15." Github. Accessed: Sep. 2021. [Online]. Available: https://github.com/vinayakumarr/NetworkIntrusionDetection/blob/master/UNSW-NB15/CNN/multiclass/cnn2.py.

[7] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. AlNemrat and S. Venkatraman, "Deep Learning Approach for Intelligent Intrusion Detection System," IEEE Access, vol. 7, pp. 41525–41550, 2019, doi: 10.1109/ACCESS.2019.2895334.

[8] J. Kevric, S. Jukic and A. Subasi, "An effective combining classifier approach using tree algorithms for network intrusion detection," Neural Comput. & Applic. vol. 28, pp. 1051—1058, 2017, doi:10.1007/s00521- 016-2418-1.

[9] N. Sultana, N. Chilamkurti, W. Peng and R. Alhadad, "Survey on SDN based network intrusion detection system using machine learning approaches," Peer-to-Peer Netw. Appl. vol. 12, no. 2, pp. 493-–501, 2019, doi:10.1007/s12083-017-0630.

[10] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A Deep Learning Approach for Network Intrusion Detection System,", in Proc. 9th EAI Int. Conf. Bioinspired Inf. Commun. Technologies (formerly BIONETICS), in BICT'15, Brussels, BEL, 2016, pp. 21—26, doi: 10.4108/eai.3-12-2015.2262516.

[11] V. Kumar. Network Intrusion Detection on UNSW-NB15.Github.Accessed:Sep.2021. [Online].Available:https://github.com/vinayakumarr/NetworkIntrusionDetection/blob/master/UNSWNB15/CNN/.

[12] A. Golrang, A.M. Golrang, S.Y. Yayilgan, A novel hybrid IDS based on modifed NSGAII-ANN and random forest.Electronics9(4),577(2020). https://doi.org/10.3390/electronics9040577

[13] M. Tang, M. Alazab, Y. Luo and M. Donlon, "Disclosure of cyber security vulnerabilities: time series modelling", Int. J. Electron. Secur. Digit. Forensics, vol. 10, no. 3, pp. 255-275, 2018.

[14] D. Larson, "Distributed denial of service attacks–holding back the flood", Netw. Secur., vol. 2016, no. 3, pp. 5-7, 2016.

[15] L. Haripriya, M.A. Jabbar, 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)" Role of Machine Learning in Intrusion Detection System: Review"

[16] Nimmy Krishnan, A. Salim, 2018 International CET Conference on Control, Communication, and Computing (IC4) "Machine Learning-Based Intrusion Detection for Virtualized Infrastructures".

[17] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, ''Toward generating a new intrusion detection dataset and intrusion traffic characterization,'' in Proc. 4th Int. Conf. Inf. Syst. Secure. Privacy, 2018, pp. 108–116, doi: 10.5220/0006639801080116.

[18] A. Ramachandran, S. Seetharaman, N. Feamster, and V. Vazirani, ''Fast monitoring of traffic subpopulations,'' in Proc. 8th ACM SIGCOMM Conf. Internet Meas. Conf. (IMC), Vouliagmeni, Greece, 2008, pp. 257–270, doi: 10.1145/1452520.1452551.

[19] J. Mai, A. Sridharan, H. Zang, and C.-N. Chuah, ''Fast filtered sampling,'' Comput. Netw., vol. 54, no. 11, pp. 1885–1898,Aug.2010, doi: 10.1016/j.comnet.2010.01.015.

[20] N. Duffield, C. Lund, and M. Thorup, ''Learn more,
sample less: Control of volume and variance in network measurement,'' IEEE Trans. Inf. Theory, vol. 51, no. 5, pp. 1756–1775, May 2005, doi: 10.1109/TIT.2005.846400.

[21] N. Hohn and D. Veitch, ''Inverting sampled traffic,'' IEEE/ACM Trans. Netw., vol. 14, no. 1, pp. 68–80, Feb. 2006, doi: 10.1109/TNET.2005.863456.

[22] P. Tune and D. Veitch, ‘‘Towards optimal sampling for flow size estimation,’’ in Proc. 8th ACM SIGCOMM Conf. Internet Meas. Conf. (IMC), 2008, pp. 243–256, doi: 10.1145/1452520.1452550.

