Deepflow: A Software-Defined Measurement System for Deep Learning

Prasanna Kumar Lakineni
Assistant Professor,
Department of CSE,
GITAM School of Technology, GITAM University, Visakhapatnam
lpk.lakineni@gmail.com

Saurabh kumar
Assistant professor
Invertis Institute of Computer Applications
Invertis university
Bareilly
saurabh.k1@invertis.org

Sanjay Modi
Lovely Professional University
Phagwara

Kapil Joshi
Uttaranchal Institute of Technology
Uttaranchal University
kapilengg0509@gmail.com
https://orcid.org/0000-0003-1097-8347

V. Mareeskannan
Asst. Professor
Dept. of CSE
St. Martin's Engineering College
Secunderabad, Telangana, India.
mareeskannan8@gmail.com

Jayapal Lande
School of Computer Science and Artificial Intelligence
SR University
Warangal, Telangana, India

2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) | 979-8-3503-9926-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICACITE57410.2023.10182469

***Abstract*-Delivering perfectly alright real - time traffic information is crucial for managing a wide range of networks, particularly vehicular communications, anomaly analysis, networking accounting, and available bandwidth. Application networking might be able to give fine-grained evaluation by offering details for each sent rules of just an Open circulation switching. Providing absolutely adequate real - time traffic information in hardware switches also poses serious problems because of the size constraints of TCAMs that can only accommodate a minimal number of rules in contrast to the number of current fluxes in the networks. Inside this editorial, we initiate Intense Flow going, a scheme for modular app assessing that's also premised on an efficient method that a) flexibly senses the channel's highest traffic references and locations prefixes, b) collects coarse-grained stream size readings for less energetic identifiers and perfectly alright metrics for the more engaged users; c) includes historical metrics to coach a cloud-based a profound learners model that has the potential to create short forecasts anytime precise f Due to the lack of the need for additional flow sampling methods that compromise accuracy, a large increase in the number of totally acceptable flows that may be recorded is now possible. . Deep Flowing can provide incredibly high accuracy for estimating flow quantities at various hierarchy levels, according to a rigorous experimental analysis using a prototype versions and actual networking signals.**

Keywords-Longer Short-Term memories (LSTM), depth training, stream pattern modeling, and congestion forecasts are some of the terms used in the study of congestion.

I. Introduction

It is crucial to be having direct exposure to perfectly alright web spine traffic measurement techniques for a variety of assignments, such as transit planning, anomaly analysis, web financial analysis, efficiently, available bandwidth, Traffic Matrix (TM) prognostication, as well as other monitoring mechanism and improvement projects. Switching that enable Free Flow has meters for every forwarding rule [1-5], showing the overall volume of bytes or packages sent. It can offer a variety of perfectly alright measurement tasks that can give insights into the operating systems, so long that a forwarding rule isn't too broad to satisfy various flows. Nevertheless, this same TCAM storing [6-10] of a switching is fundamentally limited in capacity because of its expensive cost and energy usage. The significant majority most SDN [11-13] equipment makers limit TCAMs towards the fewer then 4K L2/L3 policies since SDN-enabled switching may route hundreds of millions of streams at once. Hardware switching must implement effective methods that can actually permit more accurate measuring tasks despite the limited TCAM capabilities in addition to gain a full knowledge of the networks. Earlier research on SDN networks assessment either have focused on specific measuring objectives like big hitter identification or anomaly analysis or presumed special technology capabilities (like sketches) in the switching [14, 18].

The upper K (K of being restricted by the main parameters TCAM interior) most important flows that had been assessed with specific search regulations, and the remainder are analyzed in buildup. This is due to prior task whether in assumed that TCAMs possess sufficient functionality to fit the all noticing regulations for each of the creeks in the context of TM forecasting for SDN, that aren't the case in reality. These procedures, though, are inadequate to provide a thorough, in-depth understanding of the networks. Accomplishments: In this study, we propose Profound, a method for precise software [20-21] defined networks measures with support for predictions that can be deployed right away to gear that is already in use. Deep Flow [32-36] performs at the control and data planes using four major results. First, a particular stream (or combination of streams), identified by a source and destinations IP prefixes, can be seen in any switching that it traverses in the
and its accurate measuring site can be improved to increase the overall amount of flows analyzed. Foremost, the relatively short variability that individual connectivity tends to flow could exhibit on short time scales, CAN'17, Incheon, Republic of Korea, is scored out when interconnection streams are evaluated jointly over a period of a few seconds or longer, allowing the overall data flow to be successfully predicted by a time series forecasting model [22-24].

Fourth, by comparing the shifting patterns, we may recognize the changes and adopt more reliable measurement criteria if the network's traffic patterns significantly shift at a particular epoch and then that area of an IP area is also not accurately monitored [19] during that time. We can determine the flow tags that modified the flows with a high probability using routing data along with information port statistics (that are "open" for each time).

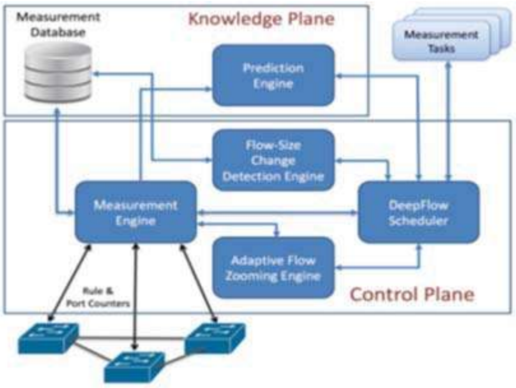

Fig. 1. Components of the Deep [29] Flow architecture.

Introduction to Deep flow Architecture The primary elements of Deep Flow are presented in this section and are seen in Figure 1. A flow has been defined as that of the communication between a source IP address as well as a network Address in the study shown below. Larger dimension, moreover, may employ the same ideas (e.g. 5-tuples). Measurement Duties: The network administrator must provide the following input data in order to begin gathering Deep Flow measurements

All flow size up to the prefixed granularity mentioned above will be taken into consideration if no limit is set. The barrier can also be expressed as a percentage of the switched' total available frequency, for example, 0.001% of the entire frequency band. Engine for Active Flow Prefix Detection: The controller's lack of a priori knowledge of the operational flow in the system and their importance is one of the primary obstacles that Deep Flow must resolve. Because to the TCAM's small volume, wild-card criteria that may match a variety of flows are frequently utilized. As a result, the precise fine-grained flows to which the flows belong cannot be determined from the traffic data of such flow. Deep Flowing uses the Dynamical Flowing Prefix Recognition Engines, that operates just on 2D IP domain that continually identifies the fastest prefixes just on system, to accomplish this. Databases Forecasting and Measurements Engine: Therefore, with order to construct a networking model, the predictions machine of Deep Stream uses historical input that has been rescued in clouds in a measurement dataset. (a supporting documentation can be utilized that varies depending on the application needs). Through an API, the Deep Flow scheduling has access to the prediction engine.

The Deep Flow planner calls the API with the destination and source IP addresses whenever a new forecast is required. After obtaining the past data from the measuring database; the predictions engine creates a forecast for the following epoch. Networking fluxes generate comparable traffic that may be grouped together by subdomain size, aim, period of night, etc., as we'll see inside the following parts. It is crucial to highlight that it isn't necessary to keep a different system for every flow combination. Engine for Flow-Size Pattern Recognition Deep Flow continuously monitors the aggregate data quantity of all the connections in order to ensure that, in the event of unexpected traffic changes within the network, the forecasting engine doesn't really generate large uncertainty. Deep Flow then uses this same Flow-Size Activity Recognition Engine to identify the prefixes that represent the cause of the volume change. In the following era, new measurement rules will be used to observe the prefixes discovered rather than making forecasts. Scheduler and Measurement Engine for Deep Flows: Given a list of measurement tasks from the network managers, a Deep Flow Scheduler determines where and when it will install precise flow data as well as which flow should employ forecasting accuracy. The Measures Engine, which handles adding or eliminating flow counters rules and retrieving flow data from TCAMs, is used by the schedule for this. The remainder fluxes are handled by Deeper Flow Scheduling using the prediction algorithm that predicts the flowing volume for the forthcoming epoch. Several steps used during optimization to enhance the quantity of flows that is being monitored are covered in the subsections which follow, where we also describe Deep Flow.

II. Estimation With Prediction

The 2 analysis techniques that Profound Flow employs are the Forecast Aided Measurement System Handset Algorithm and the Active Stream Prefix Recognition Systems, which both discover engaged stream prefixes and collect measurements for the discovered active fluid flow while interspersing them with forecasts to increase the tier of specifics of the measurement process. Proactive Flowing Prefix Detection Method, in addition of having to execute the Active Flowing Prefix Detection Method, which itself is described below, Deeper Flow remains to work in a flow ecosystem and actively detects the largest flowing within the dual IP region. This is distinct from prior efforts that assumed flows were recognized as controllers or that were given as an input with measuring tasks.

In this case, the able to detect active flows are interspersed with predictions to increase the level of detail of the measure phase, and a highest zoom-level of /32 is given. The AFPDA (Active Flow Prefix Detection Algorithm): In contrast to other efforts ([17], [16]) that presuppose the flow are known to the microcontroller or are supplied as just an input with monitor the status (e.g. [9]),

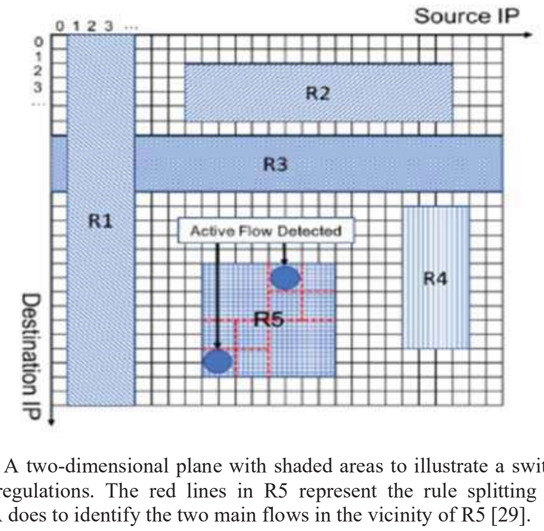

Fig. 2. A two-dimensional plane with shaded areas to illustrate a switch's set of regulations. The red lines in R5 represent the rule splitting that AFDPA does to identify the two main flows in the vicinity of R5 [29].

An Active Flow Prefix Detector (AFPDA), that is employed by Deepflow in a flow-agnostic environment, is used to proactively identify the greatest flows inside this double IP space. Once AFPDA begins, it receives all of the specifically pertaining from every switch as well as details about the network's ingress switches provided as an input network administrator during setup the next stage of AFDPA would be to begin the flowing zoom procedure, where it looks for reference IP prefix that have huge quantities and must be further divided into lengthier operational prefix which can be monitored independently.

To achieve this, AFDPA implements TCAM regulations with greater primary consideration, which, while sustaining the very same rule values as a whole, vary from the beginning rule merely in the period of the location and network Address prefixes (i.e., they were indeed effectively subgroups of each other). By shifting a portion of the matched traffic from the main rule to a different sub, it will allow for more accurate assessment. In this way, the benchmark prefix zone is gradually divided up into shorter areas until it is split once more for metrics flowing with quantities below the negligible threshold. Whenever the merged prefix duration of the desired location and origin prefixes is attained in the utter lack of a cutoff poin
the process will finish. This process is illustrated in Figure 2, for example, in which the rule storage for rule R5 is divvied up twice until 2 tends to flow (defined by the two rings) are discovered to have crossed the shield, in this case with a highest macro mode tier of /32. Prior to actually attempting to download the ability to supervise regulations for a fresh timespan, we first formulate the problem as an integer linear programming problem, something that Profound Stream tries to solve, in order to determine the best location to configure a surveillance regime while optimizing the maximum count of streams supervised. Let S = s1, s2,...,sN represent this same series of all networking devices, and let F = f1, f2, fK represent the gathering of every ((come to the appropriate place)) stream we are concerned with over a certain sampling interval.

It maximizes the overall number of laws being observed in the aforementioned linear system & ensures that we won't add many rules to a switching than it has memory for. The number of regulations per flow must equal one by force of equation. When installing the rules in a few switches in the course of each flow which ensures that the supplementary variables is binary. It represents the percentage of communication of flow that's also transmitted over physical connection, is 0 or 1.

Measurements Algorithm with Prediction (PAMA): In the beginning, Deep- Flow zooms in on flow field prefix that surpass the threshold by using the AFPDA algorithms. Deep Flow uses the PAMA method, which launches round-robin measurements once AFPDA has identified all the significant flow (the process may require more than one epoch). In order to gather d data before beginning to use forecasts for the following T epochs for every prefix, PAMA will install complete specification for the current prefixes. So, if the network's total amount of memory ism, we can cover up to so many movements before we need to start gathering data once more for the first set of prefixes. Deep Flow can combine forecasts and measures as a result, giving it a more detailed understanding of the system.

III. Analysis

. This topic of modeling changing networking period is not fresh in the relevant research. The bulk of earli
focused on estimating the overall size of various flows over several minute timescales. Those algorithms are often successful at forecasting course yes traffic matrices because they took leverage of long-term correlations to estimate how the quantity observed from an observational site would evolve in the ahead.

However, considerable work has been done to predict aggregate flow sizes over shorter time intervals, as seen in [25–27]. The scope of the paper does not include a comprehensive analysis of all prior research on the subject. Second, compared to what Deepflow attempts to represent, only aggregate congestion at the connection (port) levels was modeled in more modern examples like [28-30] over long periods (i.e. 15 minutes). Due to this, we use a new strategy in Deep Flow and, motivated by current developments in deep learning.

A recurrent neural variant known like a Long-Short-Term-Memory (LSTM) architecture has been more well-known in recent decades as a result of its success at simulating complicated time series with unknown-sized temporal lags separating significant events [21]. Utilizing self-loops allows the gradient to flow for extended periods of time without disappearing or bursting is the primary concept of LSTM. The LSTM can build up information that can be "forgot" later using the input data by combining this with the usage of a forget-gate. We believe that it is the first application of LSTM systems to the modeling of perfectly alright dynamic network patterns in short timescales. We examined actual network trace from [20] and ran Mininet simulation in order to verify the efficacy of LSTM for internet traffic modeling under varied traffic situations. We will just provide a few typical outcomes that can be utilized to validate the suggested work due to the restricted space in this publication.

Traffic Analysis by CAIDA: Unknown to the general public passive network traces from CAIDA's elevated passive sensors are included in the data in [20]

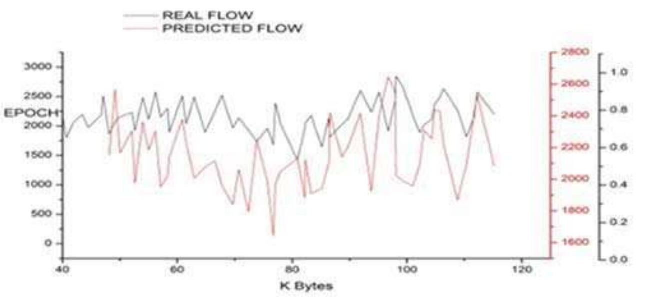

Fig. 3. A /15 aggregated flow's size is predicted by LSTM using the CAIDA trace (epoch time = 5 seconds).

Acquired in 2016 from a datacenter in Chicago, in order to simulate actual CAIDA traffic, we combine all source and destinations prefix according to the network of the specified mask size which they correspond to, which ranges from 1 and 15. We then arrange the traffic by ounce/destination IP throughout brief time intervals of one to five seconds. In th
manner, we can model traffic flow volume time - series data at different aggregate and time scales. Additionally, it was demonstrated that AFPDA converged in 16382 measures with a 1Mbps threshold, took Ninety seconds to complete and had an average precision of 14% while measuring up to 15 flows. The chart illustrates that the predictions curve rather well predicts the actual flow sizes, with such an approximate mean absolute percentage error (MAPE) =

12%. Simulations on a Mininet: With in Mininet scenario, we model the Google B4 [19] topology with 1 Gbps links & 5 hosts connected to each Open VSwitch, all of which are simultaneously transmitting traffic through their shortest paths to a chosen target host. A typical aggregate flow over 100 epochs is presented in Figure 3, along with its predictions curve and epoch size of five seconds.

Figure 4 specifically displays, for every mask size, the proportion of volume prefix (i.e., greater than the limit). The graph demonstrates that when the two - dimensional IP field is further divided, the percent decreases exponential. This illustrates that, despite the reality that there are theoretically a significant number of prefixes to investigate, AFPDA ultimately concentrates only on a small number of flows that really are crucial.

This is further demonstrated in Figure 5, which displays for every mask size the specific figure of prefixes surpassing the limit. From there, it is clear that AFPDA cumulates very quickly because it skips any micro prefix that don't meet the cutoff at mask size 8, when the number of major flows starts to decline rapidly. Last but not least, the test depicted abo had a convergence duration of 90 seconds, an era length of five seconds, and a mean of 500 measurement rules each epoch.

IV. CONNECTED WORK

There are two primary methods for measuring traffic in SDN: a) TCAM-based traffic counts (such as those in [9], [15], [16], and [17]), and [10]) random counter like sketches (e.g. [12], [13]). Since TCAM-based counters positions deplorability in commercial switches, we concentrate on the issue of traffic measuring in SDN in our work, comparable to the task in [9] and [16]. The authors of [9] put forth DREAM, a TCAM-based measuring framework that aims to strike a balance between measurement precision and the number of tools (i.e. TCAM rules) required to monitor certain flows. DREAM does not offer a general foundation for TM estimation, but it is appropriate for applications where precision can be assessed, including such (Hierarchal structure) Heavy Hitting detection and change detection. Open TM, a methodology for TM estimate for Open flow Protocol network that is built on straightforward flow data extraction from Switches, is presented by the researchers in [15].

The study makes the assumption that almost all flow rules can be tracked using targeted search rules because they all fit in the TCAM (that is no wildcard rule is used). Additionally, Open TM continuously asks flow counters from numerous switches along a flow's path, resulting in a large amount of overhead. The research examines the effectiveness of various countering retrieval tactics in its final part e.g. random vs. last switch vs. round robin etc. For TM estimate in hybrid SDN installations, the researchers of [16] suggest Open Measurement, an expansion of iStamp [11], which employs an adaptable counter insertion technique to identify significant flows and inserts exact match rules for those identified in the TCAM. The system chooses the target flow to observe based on an estimation of the magnitude of a flow during the following time frame.

The most pertinent of the earlier research from the examples above is Open Measurement [16]. However, as will be demonstrated later, there are important variations from Deep Flow. Because:

1) Open Way of measuring does not offer perfectly alright flow rate (it only selects the most significant flows that can match in the TCAM).
2) Open Way of measuring does not use projections to replace measurements; rather, it is using a model to determine the biggest flows to supervise with TCAM regulations in the following measurement epoch.
3) This same model used for assessment are simple linear designs that haven't been evaluated for their efficacy in estimating flow sizes in tiny timeframes (that is, less than 15 minutes) with a lower flow rate, where flows seem less stable,
4) Open Measure is appropriate for lengthy observation epochs with a high flow aggregate ratio observation epoch with a high flow aggregate ratio [31].

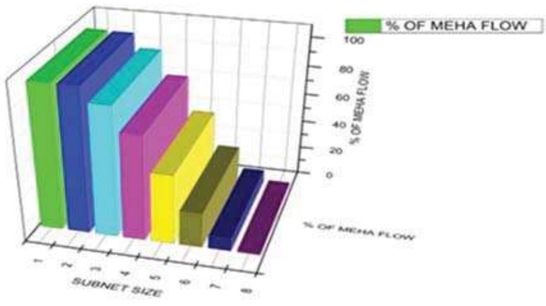

Fig. 4. The proportion of high-volume prefix, as assessed by AFPDA utilizing random traffic splitting, for every destination and source IP masks width

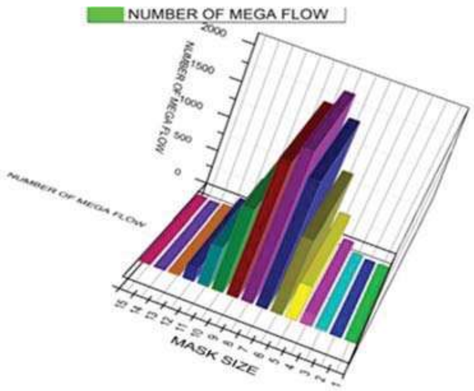

Fig. 5. The complete list of heavy frequency prefix with each intake and outgoing IP masked width as calculated by AFPDA using minor traffic partitioning.

V. FINAL COMMENTS AND FUTURE WORK

Perfectly acceptable measuring is necessary for many services and applications, so it must be offered. In this article, we propose a forecast designed to handle for SDN called Deep Flow. Deep Circulation installs complete specification for crucial flows using available TCAM recollection and uses an effective computer method to forecast this same size of a remainder flows, that can't be monitored with specific search regulations, utilizing historical time data from earlier time to evaluate. In our upcoming work, we plan to improve Deep Flow to employ complex flow dynamics and much more networking data in addition further to decrease the number of accurate flow measurements.

REFERENCES

[1] Albert Mestres, et al. 2017. Knowledge-Defined Networking.SIGCOMMComput.Commun. Rev. 47, 3 (2017), 2-10.

[2] SeppHochreiter and JurgenSchmidhuber. 1997. “Long Short-TermMemory”. Neural Comput. 9, 8 (1997), 1735-1780.

[3] Augustin Soule, KaveSalamatian, and Nina Taft. 2005. “Combiningfiltering and statistical methods for anomaly detection". In Proc. of IMC’05.USENIX, Berkeley, CA, USA, 31-31.

[4] Matthew Roughan, MikkelThorup, and Yin Zhang. 2003. “Traffic engineeringwith estimated traffic matrices". In Proceedings of the 3rd ACMSIGCOMM conference on Internet measurement (IMC ’03). ACM, NewYork, NY, USA, 248-258.

[5] K. Kumar, L. Varshney, A. Ambikapathy, K. Malik, K. Vanshika and A. Vats, "Image Denoising by Wavelet Based Thresholding Method," 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2022, pp. 63-73, doi: 10.1109/ICACITE53722.2022.9823415.

[6] Kumar K. Artificial neural network based face detection using gabor feature extraction. Int J Adv Technol Eng Res. , vol. 2, no. 4, pp. 220 -225, July 2012.

[7] CristianEstan and George Varghese. 2002. “New directions in trafficmeasurement and accounting". SIGCOMM Comput.Commun. Rev. 32,4 (August 2002), 323-336.

[8] A. Yassine, H. Rahimi and S. Shirmohammadi. 2015. “Software definednetwork traffic measurement: Current trends and challenges," in IEEEInstrumentation & Measurement Magazine, vol. 18, no. 2 (April 2015),42-50.

[9] MasoudMoshref, Minlan Yu, Ramesh Govindan, and Amin Vahdat. 2014.DREAM: dynamic resource allocation for software-defined measurementIn Proceedings of the 2014 ACM conference on SIGCOMM. ACM, New York, NY, USA, 419-430.

[10] D. Kreutz, F. M. V. Ramos, P. E. Verissimo, C. E. Rothenberg, S. Azodolmolkyand S. Uhlig, "Software-Defined Networking: A ComprehensiveSurvey," in Proceedings of the IEEE, vol. 103, no. 1, pp. 14-76, Jan. 2015.

[11] Malboubi, Mehdi, et al. "Intelligent sdn based traffic (de) aggregationand measurement paradigm (istamp)." Proceedings of IEEE INFOCOM,2014.

[12] Yu, Minlan, Lavanya Jose, and Rui Miao. "Software-Defined TrafficMeasurement with OpenSketch."Presented as part of the 10th USENIXSymposium on Networked Systems Design and Implementation (NSDI13). 2013.

[13] Moshref, Masoud, et al. "SCREAM: Sketch Resource Allocation forSoftware-defined Measurement." ACM CoNEXT, 2015.

[14] Liu, Zaoxing, et al. "One sketch to rule them all: Rethinking networkflow monitoring with UnivMon." Proceedings of the 2016 conference onACM SIGCOMM, 2016.

[15] Tootoonchian, Amin, MoniaGhobadi, and YasharGanjali. "OpenTM:traffic matrix estimator for OpenFlow networks." International Conferenceon Passive and Active Network Measurement.Springer BerlinHeidelberg, 2010.

[16] Liu, Chang, AMehdiMalboubi, and Chen-Nee Chuah. "OpenMeasure:Adaptive flow measurement & inference with online learning in SDN."Computer Communications Workshops (INFOCOM WKSHPS), IEEE,2016.

[17] Gong, Yanlei, et al. "Towards accurate online traffic matrix estimationin software-defined networks." Proceedings of the 1st ACM SIGCOMMSymposium on Software Defined Networking Research. ACM, 2015.

[18] Shilpa Choudhary, Abhishek Sharma, Shradha Gupta, HemantPurohit, SmritiSachan, “Use of RSM technology for the optimization of received signal strength for LTE signals under the influence of varying atmospheric conditions”, Transdisciplina Research and Education Center for Green Technologies, Kyushu University, Vol 7, no 4, pp. 500-509, 2020.

[19] Van Adrichem, Niels LM, Christian Doerr, and Fernando A. Kuipers."Opennetmon: Network monitoring in openflow software-defined networks."2014 IEEE Network Operations and Management Symposium(NOMS).IEEE, 2014.

[20] Jain, Sushant, et al. "B4: Experience with a globally-deployed softwaredefinedWAN." ACM SIGCOMM Computer Communication Review 43.4(2013): 3-14.

[21] Shilpa Choudhary, Abhishek Sharma, KashishSrivastava, HemantPurohit, Mudita Vats, “Read range optimization of low frequency RFID system in hostile environmental conditions by using RSM approach”,Transdisciplinary Research and Education Center for Green Technologies, Kyushu University, Vol 7, no. 3, pp. 396-403, 2020.

[22] CAIDA Anonymized Internet Traces 2016.http://www.caida.org/data/passive/passive_2016_dataset

[23] Ian Goodfellow and YoshuaBengio and Aaron Courville, "Deep Learning",MIT Press, 2016.

[24] K. Papagiannaki, K. Papagiannaki, N. Taft, N. Taft, Z. Zhang, Z. Zhang,C. Diot, and C. Diot, "Long-Term Forecasting of Internet BackboneTraffic: Observations and Initial Models", vol. 0, no. C, pp. 1178-1188,2003.

[25] K. U, Z.-L. Zhang, and S. Bhattacharyya, "Profiling internet backbonetraffic", ACM SIGCOMM Comput.Commun.Rev., vol. 35, no. 4, p. 169,2005.

[26] You, C. and Chandra, K., "Time Series Models for Internet Data Traffic",In Proc. of IEEE LCN 1999.

[27] C. Barakat, P. Thiran, G. Iannaccone, C. Diot, and P. Owezarski, "ModelingInternet backbone traffic at the flow level," IEEE Trans. SignalProcess, vol. 51, no. 8, pp. 1-12, 2003.

[28] S. Basu and A. Mukherjee, "Time Series Models for Internet Traffic",in 24th Conf. on Local Computer Networks, Oct. 1999, pp. 164-171.

[29] A. Sang and S. Li, "A Predictability Analysis of Network Traffic", inINFOCOM, Tel Aviv, Israel, Mar. 2000.

[30] A. Azzouni and G. Pujolle, "A Long Short-Term Memory RecurrentNeural Network Framework for Network Traffic Matrix Prediction",CoRR abs/1705.05690, June 2017.

[31] Lazaris, A. and Prasanna, V.K., 2017, December. DeepFlow: a deep learning framework for software-defined measurement. In Proceedings of the 2nd Workshop on Cloud-Assisted Networking (pp. 43-48).

[32] Aggelos Lazaris, and Viktor K. Prasanna "DeepFlow: A Deep Learning Framework For Software-defined Measurement"Ming Hsieh Department of Electrical Engineering University of Southern California Los Angeles, CA

[33] Ajay. P, Nagaraj. B, Ruihang Huang, "Deep Learning Techniques for Peer-to-Peer Physical Systems Based on Communication Networks", Journal of Control Science and Engineering, vol. 2022, Article ID 8013640, 12 pages, 2022. https://doi.org/10.1155/2022/8013640.

[34] J. S. Dhatterwal, M. Singh Naruka and K. S. Kaswan, "Multi-Agent System based Medical Diagnosis Using Particle Swarm Optimization in Healthcare," 2023 International Conference on Artificial Intelligence and Smart Communication (AISC), Greater Noida, India, 2023, pp. 889-893, doi: 10.1109/AISC56616.2023.10085654.

[35] K. S. Kaswan, M. S. Naruka and J. S. Dhatterwal, "Enhancing Effective Learning Capability of SOAR Agent based Episodic Memory," 2023 International Conference on Artificial Intelligence and Smart Communication (AISC), Greater Noida, India, 2023, pp. 898-902, doi: 10.1109/AISC56616.2023.10085002.

[36] DeepFlow: a deep learning framework for software-defined measurement