Research Statement

Francesco Bronzino, francesco.bronzino@ens-lyon.fr

My research focuses broadly on leveraging emergent technologies to engineer complex, deployable software systems designed to improve network service performance. Driven by dramatically increased computation capabilities, enabling new approaches such as software defined networks and radios, modern networks boast unprecedented levels of flexibility. My core expertise lies in the ability to exploit computing resources available in the network infrastructure to develop network management tools that can profoundly impact how networks operate. To achieve this goal, I pay particular attention to understanding how services naturally function on the Internet, breaking down their core components, and analyzing factors that impact their performance. This is a fundamental step towards understanding the inherent trade-offs required to accommodate new technical advances into service design and deployment strategies.

As networks become more capable of integrating flexible interfaces and ubiquitous computing components, we have the opportunity to rethink how we manage them and exploit their new capabilities to support applications and services. Advances in data science and machine learning have radically changed how we approach a number of network problems. Thanks to the commoditization of advanced data driven technologies, *e.g.*, neural networks, complex tasks such as detecting network failures or extracting information from encrypted traffic streams are now possible. Previous research, however, has primarily focused on maximizing a singular dimension—accuracy—of an offline inference pipeline, without considering the *cost* of gathering and analyzing required features that ultimately determine whether inference models are feasible *in practice*. Furthermore, the over-reliance on large datacenters to provide the necessary computing power to execute these inference tasks effectively prevents their applicability in the context of low latency close loop control applications, *e.g.*, cyber-physical and AR/VR multimedia systems, where time constraints necessitate executing them as close as possible to the edge of the network. Overcoming these limitations will enable the possibility of using machine learning techniques across a wider variety of networking problems, and allow them to be applied in practice.

My research addresses these challenges by bridging state-of-the-art concepts in networked systems and measurement together with data science and machine learning applied to network inference and management. A key aspect to my work is a al approach to advanced research informed by extensive experience converting academic research projects into functional solutions. Throughout my career, I have striven to build software that 1) validates the proposed system designs via real-world rigorous evaluation, 2) that can be verified, reused, and extended by the research community, and 3) that can achieve important broader implications for both ISPs and consumers at large.

In the near term, my research plan is to approach network systems research challenges from two angles that can be broadly summarized as: 1) addressing the challenges that limit the ability to operationalize machine learning models in the context of network management; 2) develop new network system techniques that exploit heterogeneous computing resources to support the execution of complex machine learning pipelines at the edge of the network. The following sections provide a brief overview of the research efforts I am currently involved in.

Bridging the gap between machine learning and networked systems.

Applications of machine learning to networking, from performance diagnosis to security, have conventionally relied on models that are trained on large volumes of offline traffic data. Collection of offline data for model training is normally performed in one of two ways, both falling short of providing an appealing solution. The first approach relies on collecting entire packet traces in laboratory environments, typically without regard to the model features cost or the resulting model complexity. Generated models generally do not take into account a variety of operational considerations that would be required to systematize the models, including data collection costs, feature engineering, and system data representation, that ultimately determine model performance and feasibility in practice. Alternatively, models are created using traffic features extracted through existing measurement tools deployed in operational networks. Existing collection mechanisms, such

as packet capture or flow-based monitoring (*e.g.*, IPFIX [CB08], NetFlow [CSVD04]) or more recent methods that rely on streaming algorithms [YJM13, LMV+16, CJSS03, KSXW04], are normally designed to operate in the scale of enterprises, data centers, or network backbones. For this reason their measurements are often too coarse to effectively design models that enable the analysis that could help network operators answer even basic questions about the performance of specific applications, such as video streaming, and for identifying attacks. As a result, there remains a significant gap between how statistical models for network operations are designed and their applicability in practice.

As modern networks evolve, applications of machine learning for network management and inference tasks should take advantage of the technical advances in the underlying infrastructure to overcome this gap. Specifically, modern network devices now facilitate increased flexibility through programmable hardware and increased control through software-based orchestration. These trends create a range of opportunities to more efficiently and effectively perform numerous management tasks. Network devices used in programmable operational networks have now the capability to gather diverse network traffic statistics and representations. This suggests that going forward, network management problems should be solved through a co-design approach to model design and network features collection. To achieve this, model designs must consider the range of inherent trade-offs in feature collection cost and model complexity, together with the resulting model performance for a given environment. The resulting real-time inference and prediction techniques, coupled with the increased software driven network capabilities, will create new opportunities to automate many tasks, from attack remediation to performance engineering.

Technical contributions. The problem of designing inference models that can be used in practice is clearly illustrated in the context of the inference of video streaming quality metrics (*e.g.*, video startup delay and resolution) for encrypted services without compromising users' privacy, *i.e.*, without breaking encryption. Existing methods are either 1) trained using offline traces generated in controlled laboratory settings and target a single service [MS18], making them inapplicable in practice; or 2) infer performance metrics that are easier to collect within the network, but relate poorly to actual application performance (*e.g.*, average flow throughput is a bad indicator of video resolution [MHAZ18]).

Our work [BSA+19] builds on previous state-of-the-art but extends it in several ways enabling our models to function in deployment settings where the video session traffic must be identified from all traffic traversing the observed network link and the time precision of the collected statistics is more coarse (*e.g.*, due to data aggregation). Our models outperform previous state-of-the-art while providing more fine grained results. Further, we also demonstrate that our models are al through a 16-month deployment in 66 homes. Through this study, we offer new insights about the relationships between Internet "speed" and the quality of the corresponding video streams, for a variety of services. This deployment study found that the speeds that consumers purchase from their ISPs have considerably diminishing returns with respect to video quality. As the lead author, my contributions in this work extended throughout the entire project: 1) I lead the design of the machine learning models; 2) I developed the majority of tools [vid19] used to collect the dataset [dat19] used to generate and evaluate the models; 3) I developed the measurement system used in the deployment; finally, 4) I participated in managing the deployment infrastructure.

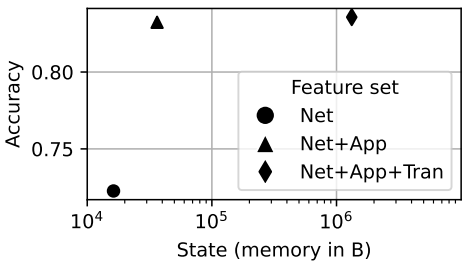

Figure 1: The relationship between features state cost and model performance for video streaming resolution inference. Maximizing model performance requires two orders of magnitude more state.

Building on the lessons learned developing models for video quality inference, we developed and released Traffic Refinery [BSA+21], a new framework and system that enables the joint evaluation of both the conventional notions of machine learning performance (*e.g.*, model accuracy) and the systems-level costs of different representations of network traffic. Traffic Refinery supports model

designers in exploring what traffic attributes may, or may not, help in modeling, while also helping them understand the implications of attempting to collect traffic features at different scales of networks. For example, we used Traffic Refinery to quantify the impact across performance and collection costs that different features had on the video quality models we had previously developed (Figure 1). Our findings using Traffic Refinery show that there is no universally ideal data representation for all network inference tasks, and thus there is value in exploring the space. Traffic Refinery lays the groundwork for new directions in applying machine learning to network traffic modeling and prediction problems, allowing operators to design machine learning pipelines that both provide accurate inferences, but are also capable of handling high traffic loads. As the lead author, my contributions in this project included the design and development of the system [tr21] (about 7k lines of go code), as well as leading the definition and evaluation of the deployment use case scenarios.

Future work. *[Line of work actively supported by the ANR-NSF MINT award in collaboration with University of Chicago and Stanford University].* Looking towards the opportunities created by modern network capabilities, my work will address a number of key challenges that currently limit the ability to employ machine learning based solutions to a larger pool of network management and inference problems in operational networks.

Traffic Refinery demonstrates the viability and utility of exploring model and traffic representations to automatically identify optimal operation points for a variety of problems. I will build on these results to develop tools that generalize and automate many of the burdensome tasks required to evaluate different models, and their underlying parameters. Starting from a given dataset provided in standard format (*e.g.*, a network trace), the new system will automatically perform feature selection, model search, and hyperparameter optimization. This system will exploit both conventional data representations as well as less explored ones (*e.g.*, temporal or inter-flow relationships) to support the analysis of a variety of supervised and unsupervised learning tasks.

Further exploiting the lessons learned in developing packet processing pipelines tailored for feature collection for machine learning problems, I will explore how to design network monitoring systems capable of collecting features that are both necessary and sufficient to inform machine learning models. I will study inherent system trade-offs, developing systems that can proactively adapt to varying traffic loads and perform *adaptive feature sampling*, balancing feature costs and anticipated model performance in real-time for multiple models. Further, I will study how to extract lightweight features using programmable network devices from different vantage points and how these could impact inference models performance.

Finally, I will explore broader problems related to the operationalization of machine learning models in Internet networks. As a first step, I will tackle the problem of concept drift for forecasting models: Over time, network traffic characteristics may shift due to 1) changes in infrastructure; 2) services accessed by users; or 3) external factors such as seasonality. Network operators need ways of determining when models become inaccurate as a result of shifts in the underlying distributions, and distinguish model inaccuracy from problems that are inherent to the network. I will work on developing new techniques to automatically mitigate drift in operational networks, providing ways to gain insights into the underlying causes of drift.

Machine learning models execution on heterogeneous edge network devices.

With the rapid growth of mobile applications and services, initiated by the worldwide explosion in smartphone adoption, and now with even more demanding scenarios such as live video analytics, cloud networks are becoming increasingly distributed. Faced by exponentially growing demand, infrastructure and service providers have responded by localizing many of their resources. In particular, providers are increasingly aiming to distribute their service points of presence (*i.e.*, processing and storage) to exploit locality and serve their clients at the edge of the network. Industry and research communities alike have embraced this approach and proposed solutions like edge clouds [Sat17] (or fog computing [BMZA12]) that can better scale and provide low delay services to real-time applications.

While edge cloud distributed solutions are conceptually simple, elegant, and offer the potential

to meet strict service requirements (*e.g.*, low latency), there are also significant technical challenges associated with moving cloud processing from a centralized architecture to a loosely coupled set of servers located at the edge of the network. One central challenge is that of distributed control: by their very nature, edge clouds are placed in multiple network domains with heterogeneous bandwidth and latency properties without a single point of control. This is particularly relevant when considering the inherent mobility properties of the devices that are often used today to consume services on the Internet. A second key challenge comes from limited and heterogeneous amounts of computational power edge systems have at their disposal. In contrast to the previous data center driven cloud model, edge clouds are often co-located with the existing network equipment and deploy limited computational resources. Hence, they are capable of hosting a limited amount of applications at any point in time and might even be insufficient for the computing requirements of intensive tasks, *e.g.*, deep learning based visual analytics. To compensate for these challenges, new techniques are required to offer cross-service infrastructure sharing and avoid that these systems become a bottleneck in the service delivery path.

Technical contributions. In previous work, I have developed architectural solutions that exploit clean separation between logical layers to more easily interconnect distributed cloud resources across network domains [BMSR20, MBS+16, SBRR15]. The culmination of this work has been the design of the Named-Object based Virtual Network architecture (NOVN) [BMSR20], a clean solution to deploy a virtual network at layer 3. NOVN exploits the general communication abstractions part of the MobilityFirst Future Internet Architecture [BNSR13] to provide a virtualization solution that offers the logical simplicity of layer 2 network virtualization while offering the flexibility to control traffic across network domains. Further, NOVN enables its users to achieve deeper control of routing policies through a technique called Application Specific Routing (ASR). ASR is particularly useful in supporting highly distributed scenarios like edge clouds where enabling network routing decisions to be made with awareness of application parameters can reduce the performance impact produced by the additional overhead generated by running such systems. Figure 2 shows an example of the ASR decision logic in the context of connecting mobile devices to the "best" edge cloud server: while in a normal networking environment "best" might correspond to the "nearest", in heterogeneous environments, varying computing loads might require delivery to a lightly loaded server which might not necessarily be the closest one in terms of network distance. My contributions to this project were manyfold: Not only I lead the design of the architecture, protocols, and routing mechanisms of NOVN, but I also was the student lead developer for the prototype used to evaluate the entire MobilityFirst architecture. The prototype included several software systems from software routers to the clients' host stack [mfc17] (about 70k lines of C++ and Java code).

To further understand the impact of system heterogeneity, we analytically studied how key design parameters affect scalability and performance of edge cloud deployments [MBRS18]. Using a custom model we developed, we answered a number of questions spanning from how much load can an edge cloud network support without affecting the performance of an application, to what impact inter-edge (fronthaul) and edge-to-core (backhaul) network bandwidths have on the system capacity. The results provided insights and guidance on how to select the right balance between edge and core cloud resources for delay constrained applications. In this project, I worked with the lead author, a junior Ph.D. student, in developing the model design and evaluation.

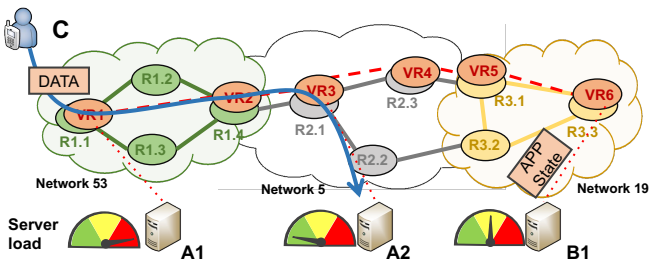

Figure 2: NOVN mechanisms enable anycast routing across service instances based on network metrics, *e.g.*, distance, and application specific metrics, *e.g.*, server load.

Future work. *[Line of work actively supported by the ANR PARFAIT award in collaboration with Avignon Université, Inria, and CNAM].* Moving forward, my work will focus on how to address

the joint problem of executing diverse machine learning pipelines on heterogeneous computing resources at the edge of the network. An effective example of this problem is represented by live video analytics. Driven by the advances of neural network based video feeds processing and the advent of edge computing, live video analytics have established themselves as a core network application of the future [ABB+17]. Existing solutions have focused largely on fixed cameras whose video streams traverse a hierarchy of clusters and a somewhat predictable path in the network to execute vision pipelines [HAB+18, JAB+18, ZLL+19]. With the easier-than-ever access to and rising penetration of mobile cameras and devices, there is a need to include these feeds into the design of the analytics architecture. Mobile video sources include, but are not limited to, smartphones and drones, as well as augmented reality visors that require live processing based on a specific viewpoint. As shown by our preliminary work, in the absence of real-time support for gathering analytics from mobile source, we are missing an opportunity for truly pervasive intelligent analytics [FBDP21, RBJ21]. The advances of novel machine learning techniques together with the increased accessibility in edge computing resources provide the perfect platform to integrate these devices into intelligent analytics pipelines.

I will attack the challenges raised by these scenarios both through the integration of all compute resources available at the edge as well as by advancing the design of efficient machine learning methods for edge execution. First, I will leverage concepts from network function virtualization to enable dynamic chaining of modules in video analytics pipeline across a distributed network. While performance and scalability will remain key, I will investigate other issues such as efficient computation offload, end-device battery limitations, and bandwidth management. Further, I will work towards the integration of low cost GPU/TPU based single board devices [jet20, cor20] into the processing pipeline. To overcome their inherent computing limitations, I will investigate how to integrate lightweight machine learning models (*e.g.*, through model distillation [HVD15] or regularization and pruning [TLFF18]) into the processing pipeline, analyzing the trade-offs of combining them with more complex models that can be executed on more capable hardware. Finally, to bring all the components together, I will work on defining new scheduling algorithms rooted in Markov decision and learning theory that dynamically take into account the available heterogeneous resources to coordinate processing and maximizing the total performance of the system. In this context, my solutions will aim to solve the challenge of taking decisions under time-varying service demands and technological constraints while facing partial and possibly asynchronous information on the performance of applications' deployed in the network.

Conclusions.

Over the years my research has had significant academic and social impact. My work has been funded by national and international funding agencies (two active awards as PI [MIN, PAR]), has been published in premier venues in network systems research (*e.g.*, [BSA+21, LSBF21, BSA+19, BMSR20, MBTS19, MBRS18, SBRR15]), and was awarded two best paper awards as well as the Rutgers ECE Graduate Program Academic Achievement Award. My research on the impact of residential broadband speeds on video streaming quality was highlighted on the front page of *The Wall Street Journal* [wsj19]. Further, the software produced during my research work was both used within the experimental research community, in particular within the GENI nationwide testbed initiative [gen], and adopted for technology transfers to industrial partners. For example, the MobilityFirst architecture prototype [mfc17] was used as the foundation to develop and deploy a working proof-of-concept satellite-based video distribution network in collaboration with SES in Princeton [ses16]. Finally, the industrial experience acquired during my time at Nokia Bell Labs, as well as my internship experiences done during graduate school, have been key in establishing fruitful partnerships with industry partners that have been and will be beneficial in my future endeavors.

I believe my research goals, largely focused on operationalizing machine learning models in the context of network management and systems, closely align with the ongoing evolution of the Laboratoire de l'Informatique du Parallélisme at ENS Lyon and its mission to develop top research. My program is a natural fit with the HoWNet team. The group is already actively working on machine learning driven network management solutions in the context of wireless networks (*e.g.*, [BBB21]). My research approach could directly integrate exiting projects (*e.g.*, Wi-Fi resource management

based on application quality inference) as well as extend them to the context of wired operational networks. Further, distributed approaches to machine learning could benefit ongoing work on communications optimization (*e.g.*, [GLS22]). Finally, my wide experience in converting research projects into experimental realizations, both using research testbeds and in the wild, will be pivotal in incrementing the groups efforts in this space, particularly in the context of networking systems performance evaluation.

Beyond the HoWNet team, my program is well positioned to enable lasting research collaborations within the LIP. First, my expertise managing and working with research testbeds (*e.g.*, ORBIT and GENI) and evaluating computing systems well matches the ongoing efforts of the AVALON team who is leading the development of major testbeds in France and Europe (*e.g.*, Grid'5000 before, SLICES currently). Second, my work on executing machine learning pipelines on heterogeneous distributed computing nodes could greatly benefit from interactions with the ROMA team. To maximize the potential of these systems, we need to develop advanced multi-criteria scheduling strategies that taking into consideration the availability of diverse hardware accelerators (*i.e.*, CPUs / GPUs / TPUs) while meeting performance requirements and timing constraints. Finally, my work on machine learning applied to networking well pairs with the expertise of the DANTE team and the formal machine learning methods at the center of their work. In particular, looking at how to formally represent network traffic and network structures to maximize the effectiveness of the management techniques developed.

References.

[ABB+17] Ganesh Ananthanarayanan, Paramvir Bahl, Peter Bodík, Krishna Chintalapudi, Matthai Philipose, Lenin Ravindranath, and Sudipta Sinha. Real-time video analytics: The killer app for edge computing. *computer*, 50(10):58–67, 2017.

[BBB21] Anthony Bardou, Thomas Begin, and Anthony Busson. Improving the spatial reuse in ieee 802.11 ax wlans: A multi-armed bandit approach. In *Proceedings of the 24th International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 135–144, 2021.

[BMSR20] Francesco Bronzino, Sumit Maheshwari, Ivan Seskar, and Dipankar Raychaudhuri. Novn: A named-object based virtual network architecture to support advanced mobile edge computing services. *Pervasive and Mobile Computing*, 69:101261, 2020.

[BMZA12] Flavio Bonomi, Rodolfo Milito, Jiang Zhu, and Sateesh Addepalli. Fog computing and its role in the internet of things. In *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*, pages 13–16. ACM, 2012.

[BNSR13] Francesco Bronzino, Kiran Nagaraja, Ivan Seskar, and Dipankar Raychaudhuri. Network service abstractions for a mobility-centric future internet architecture. In *Proceedings of the eighth ACM international workshop on Mobility in the evolving internet architecture*, pages 5–10. ACM, 2013.

[BSA+19] Francesco Bronzino, Paul Schmitt, Sara Ayoubi, Guilherme Martins, Renata Teixeira, and Nick Feamster. Inferring streaming video quality from encrypted traffic: Practical models and deployment experience. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(3), December 2019.

[BSA+21] Francesco Bronzino, Paul Schmitt, Sara Ayoubi, Hyojoon Kim, Renata Teixeira, and Nick Feamster. Traffic refinery: Cost-aware data representation for machine learning on network traffic. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 5(3):1–24, 2021.

[CB08] Benoit Claise and Stewart Bryant. Specification of the ip flow information export (ipfix) protocol for the exchange of ip traffic flow information. Technical report, RFC 5101, January, 2008.

[CJSS03] Chuck Cranor, Theodore Johnson, Oliver Spataschek, and Vladislav Shkapenyuk. Gigascope: a stream database for network applications. In *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 647–651. ACM, 2003.

[cor20] Coral Dev Board. `https://coral.ai/products/dev-board/`, 2020.

[CSVD04] Benoit Claise, Ganesh Sadasivan, Vamsi Valluri, and Martin Djernaes. Cisco systems netflow services export version 9. 2004.

[dat19] Labeled video sessions dataset. `https://nm-public-data.s3.us-east-2.amazonaws.com/dataset/all_traffic_time_10.pkl`, 2019.

[FBDP21] Francescomaria Faticanti, Francesco Bronzino, and Francesco De Pellegrini. The case for admission control of mobile cameras into the live video analytics pipeline. In *Proceedings of the 3rd ACM Workshop on Hot Topics in Video Analytics and Intelligent Edges*, pages 25–30, 2021.

[gen] Deploying Future Internet Applications in Mobility First Project's GENI-Based Environment. `https://bit.ly/31wCG2o`.

[GLS22] Rémy Grünblatt, Isabelle Guérin Lassous, and Olivier Simonin. A distributed antenna orientation solution for optimizing communications in a fleet of uavs. *Computer Communications*, 181:102–115, 2022.

[HAB+18] Chien-Chun Hung, Ganesh Ananthanarayanan, Peter Bodik, Leana Golubchik, Minlan Yu, Paramvir Bahl, and Matthai Philipose. Videoedge: Processing camera streams using hierarchical clusters. In *2018 IEEE/ACM Symposium on Edge Computing (SEC)*, pages 115–131. IEEE, 2018.

[HVD15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[JAB+18] Junchen Jiang, Ganesh Ananthanarayanan, Peter Bodik, Siddhartha Sen, and Ion Stoica. Chameleon: scalable adaptation of video analytics. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, pages 253–266. ACM, 2018.

[jet20] Jetson Nano Developer Kit. `https://developer.nvidia.com/embedded/jetson-nano-developer-kit`, 2020.

[KSXW04] Abhishek Kumar, Minho Sung, Jun Jim Xu, and Jia Wang. Data streaming algorithms for efficient and accurate estimation of flow size distribution. In *ACM SIGMETRICS Performance Evaluation Review*, volume 32, pages 177–188. ACM, 2004.

[LMV+16] Zaoxing Liu, Antonis Manousis, Gregory Vorsanger, Vyas Sekar, and Vladimir Braverman. One sketch to rule them all: Rethinking network flow monitoring with univmon. In *Proceedings of the 2016 ACM SIGCOMM Conference*, pages 101–114. ACM, 2016.

[LSBF21] Shinan Liu, Paul Schmitt, Francesco Bronzino, and Nick Feamster. Characterizing service provider response to the covid-19 pandemic in the united states. In *PAM 2021-Passive and Active Measurement Conference*, 2021.

[MBRS18] Sumit Maheshwari, Francesco Bronzino, Dipankar Raychaudhuri, and Ivan Seskar. Scalability and performance evaluation of edge cloud systems for latency constrained applications. In *2018 IEEE/ACM Symposium on Edge Computing (SEC)*, pages 286–299. ACM, 2018.

[MBS+16] Shreyasee Mukherjee, Francesco Bronzino, Suja Srinivasan, Jiachen Chen, and Dipankar Raychaudhuri. Achieving scalable push multicast services using global name resolution, 2016.

[MBTS19] Ivan Morandi, Francesco Bronzino, Renata Teixeira, and Srikanth Sundaresan. Service traceroute: Tracing paths of application flows. In *International Conference on Passive and Active Network Measurement*, pages 116–128. Springer, 2019.

[mfc17] Mobilityfirst fia protocol suite. https://mobilityfirst.orbit-lab.org/wiki/, 2017.

[MHAZ18] Tarun Mangla, Emir Halepovic, Mostafa Ammar, and Ellen Zegura. emimic: Estimating http-based video qoe metrics from encrypted network traffic. In *Network Traffic Measurement and Analysis Conference (TMA)*, Vienna, Austria, June 2018.

[MIN] Anr-nsf mint: Modeling modern network traffic: From data representation to automated machine learning. https://anr.fr/fileadmin/aap/2021/selection/aapg-2021-selection-PRCI-NSF-num.pdf.

[MS18] M. Hammad Mazhar and Zubair Shafiq. Real-time video quality of experience monitoring for HTTPS and QUIC. In *INFOCOM*, Honolulu, HI, April 2018.

[PAR] Anr parfait: Planning and learning for ai-edge computing. https://anr.fr/fileadmin/aap/2021/selection/aapg-selection-2021.pdf.

[RBJ21] Sri Pramodh Rachuri, Francesco Bronzino, and Shubham Jain. Decentralized modular architecture for live video analytics at the edge. In *Proceedings of the 3rd ACM Workshop on Hot Topics in Video Analytics and Intelligent Edges*, pages 13–18, 2021.

[Sat17] Mahadev Satyanarayanan. The emergence of edge computing. *Computer*, 50(1):30–39, 2017.

[SBRR15] Kai Su, Francesco Bronzino, KK Ramakrishnan, and Dipankar Raychaudhuri. Mftp: A clean-slate transport protocol for the information centric mobilityfirst network. In *Proceedings of the 2nd International Conference on Information-Centric Networking (ICN)*, pages 127–136. ACM, 2015.

[ses16] SES and Rutgers University test satellite content delivery network for sTreaming, OTT, and 5G. https://goo.gl/CjdmRy, 2016.

[TLFF18] Enzo Tartaglione, Skjalg Lepsøy, Attilio Fiandrotti, and Gianluca Francini. Learning sparse neural networks via sensitivity-driven regularization. In *Advances in neural information processing systems*, pages 3878–3888, 2018.

[tr21] Traffic refinery. https://traffic-refinery.github.io, 2021.

[vid19] Video Collection Tools. https://github.com/inria-muse/video_collection, 2019.

[wsj19] The Truth About Faster Internet: It's Not Worth It. https://www.wsj.com/graphics/faster-internet-not-worth-it/, 2019.

[YJM13] Minlan Yu, Lavanya Jose, and Rui Miao. Software defined traffic measurement with opensketch. In *Presented as part of the 10th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 13)*, pages 29–42, 2013.

[ZLL+19] Wuyang Zhang, Sugang Li, Luyang Liu, Zhenhua Jia, Yanyong Zhang, and Dipankar Raychaudhuri. Hetero-edge: Orchestration of real-time vision applications on heterogeneous edge clouds. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 1270–1278. IEEE, 2019.