SEMANTIC

end-to-end Slicing and data-drivEn autoMAtion of Next generation cellular neTworks with mobIle edge Clouds

Marie Skłodowska-Curie Actions (MSCA)
Innovative Training Networks (ITN)
H2020-MSCA-ITN-2019
861165 – SEMANTIC

WP4 – Inter Slice management and joint allocation of resources in MEC/RAN clouds

D4.1: SoA on Network Control and Automation Tools

Contractual Date of Delivery:	M18
Actual Date of Delivery:	30/06/2021
Responsible Beneficiary:	Telenor Sverige AB
Contributing beneficiaries:	TLN, IQU, NOK, EUR
Security:	Public
Nature:	Report
Version:	2.0

Document Information

Version Date: 29/07/2021
Total Number of Pages: 40

Authors

Name	Organization	Email
Qurat ulain	TLN	Qurat.ulain@telenor.se
Sudharshan Paindi Jayakumar	NOK	sudharshan.paindi_jayakumar@nokia.com
Ali Ehsnain	EUR	Ali.Ehsanian@eurecom.fr
Shahwaiz Ur Rehman	IQU	s.rehman@iquadrat.com

Document History

Revision	Date	Modification	Contact Person
V1.0	30/03/2021	Defining a preliminary table of contents (TLN)	Dr.Rius I Riu Jaume
V1.1	25/05/2021	Tools and Proposed Methodologies by TLN & Nokia	Qurat ulain & Sudharshan Paindi Jayakumar
V1.2	07/06/2021	Merging contributions of TLN & Nokia ESR	Qurat ulain & Sudharshan Paindi Jayakumar
V1.3	29/07/2021	Merging contributions of TLN, Nokia & Eurecom ESR	Qurat ulain, Sudharshan Paindi Jayakumar & Ali Ehsanian
V2.0	30/07/2021	Final documentation	Qurat ulain, Sudharshan Paindi Jayakumar & Ali Ehsanian

Table of Contents

List of Acronyms and Abbreviations

Acronym	Description
3D-DefCNN	3D Deformable Convolutional Neural Networks
ADAM	Alternating Direction Method of Multipliers
AI	Artificial Intelligence
ANN	Artificial Neural Network
AWS	Amazon Web Services
BB	Black Box
BS	Base Station
CMDP	Constrained Markov Decision Process
CNN	Convolutional Neural Network
CP	Content Provider
CSP	Communications Service Providers
DC	Data Center
DDNN	Distributed Deep Neural Network
DDPG	Deep Deterministic Policy Gradient
DL	Deep Learning
DNN	Deep Neural Network
DRL	Deep Reinforcement Learning
E2ENS	End to End Network Slicing
EFNS	Edge/Fog Network Slicing
eMBB	Enhanced Mobile Broadband
EU	User Equipment
InP	Infrastructure Provider
IPO	Interior-point Policy Optimization
KPI	Key Performance Indicator
LAO	Learning-Assisted Optimization
MANO	Management and Orchestration
MEC	Multi-access Edge Computing
ML	Machine Learning
MLP	Multi-Layer Perceptron
MMTC	Massive Machine-Type Communications
MTD	Mobile traffic Decomposition
NEs	Network Elements
NFV	Network Function Virtualization
NMS	Network Management System
NUM	Network Utility Maximization
O-RAN	Open Radio Access Network
PAAS	Platform as A Service
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RIC	RAN Intelligent Controller
RL	Reinforcement Learning
RNN	Recurrent Neural Network
SEP	Service Enablement Platform
SLA	Service Level Agreement

SNMP	Simple Network Management Protocol
URLLC	Ultra Reliable and Low-Latency Communication
VM	Virtual Machine

List of Figures

Executive Summary

SEMANTIC "end-to-end Slicing and data-drivEn autoMAtion of Next generation cellular neTworks with mobIle edge Clouds" is a H2020 ITN project funded by the EU, which aims to create an innovative research and training network for multi-GHz spectrum communications, MEC-empowered service provisioning and end-to-end network slicing, all integrated and jointly orchestrated by forward-looking data-driven network control and automation exploiting the enormous amounts of mobile big data spurred into the mobile data network.

In this context, SEMANTIC ESRs, guided by their experienced supervisors contributed to deliverable D4.1 entitled "SoA on Network Control and Automation Tools " towards the objectives of WP4 (Inter Slice management and joint allocation of resources in MEC/RAN clouds). This document summarizes the key findings of the ESRs towards the task 4.1 will serve as basis for defining the common architecture that will help them in achieving their SEMANTIC KPIs, identifying available network tools and the proposed methods including ML and DL for validating the datasets. In addition, current scientific methods and industrial tools related to network control and automation are also discussed.

Section 1: Introduction

Considering SEMANTIC use case for data-driven network control and automation, our aim is to apply data-driven mechanisms and tools to achieve the SEMANTIC KPI and target value of “10-fold improvement of the capability to re-allocate resources compared to today (time required and size of pooled resources)” [1].

Our aim is to make network more automated and intelligent, steering the network with KPIs, to achieve self-configuration, self-healing, and self-optimization. The development of data-driven processes and tools will help to translate real-time analytics into actionable insights, improve the overall network performance and achieve the abovementioned SEMANTIC KPI and target value.

Telenor (TLN) proposes the concept of Zero Touch Operation Enablement to describe how is intended to use data-driven network control and automation tools in Telenor network, see Figure 1 below.

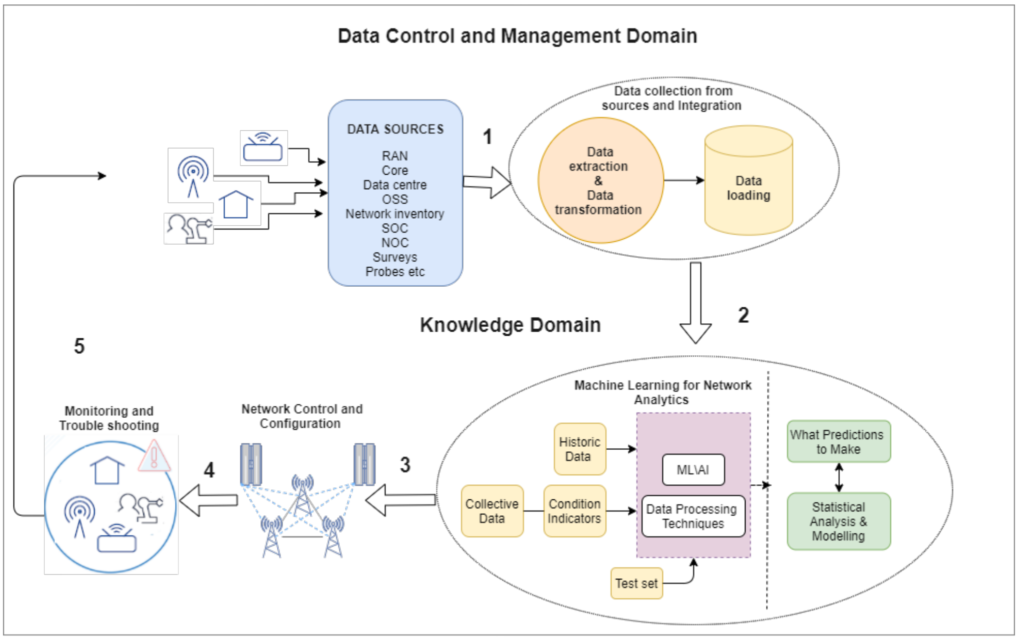

Figure 1 Zero Touch Operation Enablement concept, based on the use of data-driven network control and automation processes and tools

The telecom industry consists of a complex set of domains and subdomains. In reference to the different network domains Nokia, aims to automate, optimize the deployment and maintenance of ML systems in production for the telecommunication industry. The idea is to define a reusable yet flexible framework for automating ML/AI use case through xApps and rApps inorder to support composable pipelines over heterogeneous development and deployment conditions (such as e.g. RAN and MEC), and flexible enough to constantly integrate best-of-breed tools and solutions from open source community.
In order to deliver a viable, flexible and dynamic solution for the technological advancement, we strive to:

- Simplify data scientist life by automating the majority of repetitive tasks
- Easily replicate deployment and operation patterns/workflows across different use cases

- Automate versioning, retraining options, conversions, alerts
- Faster reaction, e.g. predictive/proactive retraining

One example of state-of-the-art tool which we incorporate as our baseline on which we build upon other tools and improvise is O-RAN SC RAN Intelligent Controller (RIC) platform [55]. High level overview of the O-RAN Alliance ML cycle can be found in Fig 2.

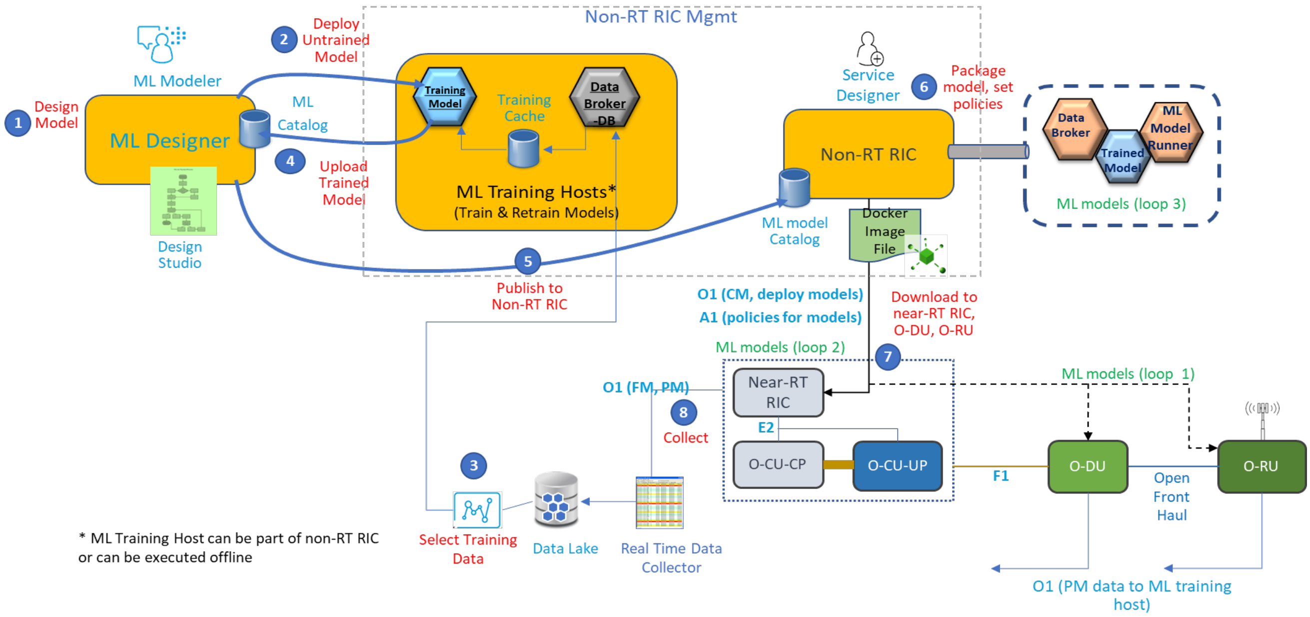

Figure 2. ML Control Loops that influence the functioning of the RIC [55]

Telenor and nokia enables, maintains, and operates services to its customers across each of its domains. They are controlling the infrastructure that their users are frequently using to access services. Due to large customer base, there is large amount and variety of data, it becomes difficult to be able to handle and analyze this data. To address this challenge, Telenor focuses on finding potential use cases across different network domains, enabling them to know where their customers are, how they interact with the network and quality pf service they experienced. This document is designed to offer tools in each network domain, as well as the proposed methods used to verify the usefulness of network data. This document mainly comprises of two sections:

1. Relevant Network Domains and their tools
   a. Radio Access Network (RAN) domain tools
   b. Transmission domain tools
   c. Core network domain tools
   d. Data center domain tools

2. Proposed Methodologies
   a. Network Crowdsource Measurement
   b. Network Performance Control
   c. Network Disturbance Control

Following the incentives set above, this document elaborates the tools identified in different network domain for data collection and proposed methodologies and concludes its relevance to SEMANTIC architecture, its integration with network performance and automation.

Section 2: State of the Art Tools on Network control and Automation

2.1. Network Domains and Tools

The semantic goal is to improve their network performance by 10 folds and this is possible only by gaining control over the network. However, it is difficult to determine whether a network performance is considered a good or bad experience by the user. To address this challenge, Telenor developed different use cases and then tested these hypotheses by collecting data from different network domains. As a result of their successful experiments, Telenor has proposed various tools that are already implemented and provide data across different network domains. The information collected from these network domains includes mobile phone internet data usage, connected cell sites, network status (e.g., download rate, reset packets, throughput), complaint history and anomalies recorded in the network etc.

2.2. RAN Domain tools

The process of evaluating the network performance, requires measurements but these measurements are complex and time consuming. But Telenor worked to track the performance of network through those KPI that can be easily accessed from their RAN domain. This is reflected in section 3.1. and 3.2. where they have looked precisely into the area of quality of experience QoE. Their tests rely on measurements collected from RAN domain, which are directly linked to the usage activity by the end user. Through the interaction between crowd source tools and network-based tools, Telenor is able to identify various features that are considered important, such as session throughput, signal strength, referral efforts, and so on. Some of the main tools that help us in carrying out our analysis on the RAN domain, besides the RAN vendors network management systems (NMS) tools, are:

- **Mycom**
  It is a performance management tool that continuously collects and monitors the radio access network performance, capacity and quality. Telenor Sweden is using Mycom for monitoring their radio KPIs [2].

- **Subtonomy**
  It's a service monitoring software that has an ability to handle, monitor and manage network transactions in real-time for each day. Telenor Sweden is using Subtonomy to collect the information about their KPI's for network traffic on cell to user equipment level including calls, data volume and handover attempts [3].

- **Home WIFI**
  Nokia's home WIFI is used to provide network throughout the homes. It has an ability of intelligent channel selection to ensure the optimal Wi-Fi channel selection, to avoid any Wi-Fi glitches. Telenor using this Wi-Fi for providing the IPTV services to its customers and able to collect information about their KPIs related to available bandwidth, physical rate, channel number and width, broadcasts, discarded and error packets, received and transmitted kilobytes, multicast or unicast packets, Wi-Fi frequency bands. It will be discussed in section 3.1.3.

- **Tutela**
  It is an independent crowdsource data that provides network measurements beyond the network layer. It offers performance measurements at application

and end user level to determine quality of experience. It covers over 300 million smartphone users. Telenor Sweden is working on this data to understand the network performance from end user level.

2.3. Transmission domain tools

The development of a monitoring system for telecommunication nodes is one of the key areas where Telenor has focused on and implemented various tools. To understand the anomalies in the network as reflected in section 3.3. the devices involved in the connection, monitoring and alarming system, its operation and other components are considered relevant for our proposed use case. Telenor uses both network protocols and dedicated probes to gain insights and data from its transmission domain. Some relevant examples follow below:

- **Polystar OSIX**
  It can be used for real-time network data collection, tracing and troubleshooting it helps communication service providers to improve quality of service by detecting and resolving network issues. Telenor Sweden is using its probes for control plane monitoring. [4]

- **Splunk**
  It is a scalable data platform that can be used for investigating, monitoring, analyzing and acting. It has machine learning capabilities for identifying, predicting and self-healing. It is designed to facilitate sudden or unexpected bursts in data volume that can be scaled in cloud. It ingests any kind of data (at rest or streaming) regardless of their source and type. [5]

- **SNMP**
  It is a simple network management protocol (SNMP) that is used in application layer for exchanging management information between different network devices. It is widely accepted by network protocols to manage and monitor network elements. [6]

- **Juniper Paragon**
  It uses active traffic to verify application and service performance at the time of service delivery and throughout the life of the service. Telenor Sweden is using its probes for collecting and monitoring the services.[7]

2.4. CORE Domain tools

Telenor is using both network protocols, such as netflow, s-flow, snmp, radius, etc., core network equipment NMS, and dedicated probes to gain insights and data from the different Core network domains. Some relevant examples follow below:

- **Netbox**
  It is a network management inventory automation tool. It is designed for IP address management and data center infrastructure management. It provides a domain-specific data source for network operations. [8]

- **Juniper Paragon:**
  It is a network monitoring and planning tool that provides in-depth network views, health audits, and power to control and monitor the traffic routing across the network in real time. Telenor Sweden is using the probe system for the core network monitoring. [7]

- **Saltstack**

It is configuration management python-based, open-source software that is used for event-driven IT automations and remote task executions. [9]

- **CISCO NSO**
  It is a model-based programmatic interface that allows network orchestration, configuration and management. [10]

- **Netscout Arbor DDoS**
  It's provides network protection and network visibility solution. It protects from DDoS attacks by providing a fully integrated, incloud and on-premise products and services, backed by continuous global threat intelligence. Telenor Sweden using this for the security and monitoring of their core. [11]

2.5. Data center domain (Cloud) tools

Telenor is using both public data center infrastructure (e.g.: Amazon Web Services -AWS-), private data centers (on premise data center infrastructure deployed and operated by Telenor) and a mix of approached between public cloud, private cloud and *hybrid approaches* such as Platform as a Service (PAAS).
For all these approaches, tools for configuration, monitoring, data collection, analysis troubleshooting are in place. Some relevant examples follow below:

- **Red Hat Ansible**
  It is an automation and scheduling platform for apps and cloud infrastructure. It provides application deployment, configuration management and continuous delivery. [12]

- **Teradata**
  It is a data ingestion tool that supports scalable high-performance data integration solutions. Having a capability to load and transform terabytes of data. [13]

- **Kafka by Apache**
  It is an open-source software bus that uses stream processing and provides unified, high-throughput, low-latency platform for handling real-time data feeds. It is written in Scala and Java. [14]

- **Elastic search**
  It is a distributed, free and open search analytics engine for all types of data, including textual, numerical, geospatial, structured, and unstructured. It is known for its simple REST APIs, distributed nature, speed, and scalability. [15]

- **Kinesis by Amazon**
  It is scalable and durable real-time data streaming service that can continuously capture gigabytes of data. [16]

- **Google analytics**
  It is used for web analytics for analyzing network traffic and user behavior. It is a super versatile tool that can also be used for SaaS Analytics. [17]

- **Red Hat OpenShift**

It is used to automate updates, scale and provision by using the enterprise scale and security of IBM cloud. It has the resiliency to handle unexpected surges. [18]

- **Jaeger**
  It is an open source software for tracing transactions between distributed services. It is used for monitoring and troubleshooting complex microservices environments and its preferred deployment method is Kubernetes. [19]

2.6. Cloud and Solution Focused Tools

Nokia product provisioning must be able implement and productize variants of the below mentioned components so that they can be deployed to different required serving environments. The exact KPI requirement is provision of new services in sub-minute scale. It is expected that:

- The Online ML Controller and the Online Training System are implemented as cloud native microservices
- The Online Inference System is implemented depending on the need of the use case to run as cloud native services or as microservices/modules for the legacy products, according to the architecture of legacy product in case (e.g. bare metal BTS)

2.6.1. Cloud Focused Tools

Below are some of the major state-of-the art cloud focused tools identified which helps in network control and automation:

- **AirFrame Open Edge Server [56]**
  x86 solution built and tailored to fully support edge and far-edge cloud deployments. The ultra-small footprint provided by the solution is complemented with a real-time, OPNFV compatible, OpenStack distribution built to provide the performance and low latency required by solutions like Cloud RAN.

- **AirScale Cloud RAN [57]**
  Together with virtualization and network automation, it is expected to result in significant improvements in the way Communications Service Providers (CSPs) manage their networks and deliver new services. Furthermore, it fosters innovation by supporting the introduction of advanced capabilities such as artificial intelligence (AI) and machine learning (ML).

- **Ansible [58]**
  Underlay VM automated deployment. Automates cloud provisioning, configuration management, application deployment, intra-service orchestration.

- **CloudBand [59]**
  Deploy cloud infrastructure, management and orchestration for Virtual Machines and Container-based software.

- **Deepfield Cloud Intelligence [60]**
  Deepfield Cloud Intelligence gives us cost-effective, real-time and actionable insight into all our network flows. This information can be used to optimize the network infrastructure and resources.

- **Logstack [61]**
  Centralized log collection and enrichment solution monitoring cloud from Openstack platform perspective. Log collection, enrichment and dashboards implemented with Elastic product family called Elastic Stack (Metricbeat, Elasticsearch, Logstash and Kibana).

- **Nessus [62]**
  Security Tool – Application Vulnerability Scanner

- **NetGuard XDR Security Operations [63]**
  NetGuard XDR provides CSPs with stronger network defenses that rapidly prevent and stop threats before they materialize. The platform modules come with new analytics, machine learning, and automation functions to better manage incidents and react faster to neutralize threats.

- **OpenStack Private Cloud for Large Enterprises [64]**
  Private cloud architecture addresses two key use cases for OpenStack: scaling OpenStack up to Webscale volumes and leveraging OpenStack in complex global multi-datacenter, hybrid cloud environments.

- **Zabbix (separate VM) [65]**
  Real-time performance monitoring for cloud infra. Can be used for monitoring cloud resources, VM traffic, various equipment via SNMP queries, alarms generation for preventive/corrective actions, customized reports generation based on collected data (capacity, performance, etc.)

2.6.2. Product and Solution Oriented Tools

Below are some of the major state-of-the art product and solution oriented tools identified which helps in network control and automation:

- **AirFrame Data Center Solution [66]**
  Provides solution from far edge to central data centers with common automation and managements system with needed workflow tools.

- **CMTool [67]**
  Tool to pull automatically the configuration from all Network Elements (NEs) in the solution and store it in a centralized repository. Automatically push configuration data to all PODs. Automatically compare the real-time configuration of the NEs with the reference configuration from the repository

- **Flowone [68]**
  E2E Service Lifecycle Orchestration of complex hybrid services including virtualized cloud resources and traditional physical resources Freedom from vendor lock-in with a multi-vendor, multi-technology and multi-domain platform

- **Deepfield Defender [69]**
  Deepfield Defender uses big data analytics to detect distributed denial of service (DDoS) threats in real time. It allows us to stop relying on myopic, sampled-and-aggregated views of the network, collected at specific network interfaces.

- **Deepfield Operational Intelligence [70]**
  Deepfield Operational Intelligence increases the operational agility and efficiency by equipping with actionable, real-time insight into network trends, including tracking of network and service deviations and anomalies.

- **NetAct [71]**
  Offers best-in-class applications for seamless daily network operations, including configuration management, monitoring and software management. NetAct supports network elements in mobile radio and core, Wi-Fi, IoT, public safety and telco cloud.

- **Nokia IPBox [72]**
  IP Networking Design and Management according to defined data model, covering underlay/overlay IP plan, inter-VNF and legacy interconnection (L3 VRFs, zones, BGP peering)

- **Release Management Tool (PowerBI + DB) [73]**
  Planning of solution releases and evolution over time, with dynamic SW compatibility visualization within each planned release

- **TEMS Pocket (Infovista) [74]**
  Vulnerabilities scan analysis (how to FROM 'a scan by whatever scan tool (NESSUS, Anchore)' TO 'a dashboard' with false positive, new vulnerabilities, vulnerabilities with correction plan, vulnerabilities w/o correction plan)

- **VulCAN [75]**
  Vulnerabilities scan analysis (how to FROM 'a scan by whatever scan tool (NESSUS, Anchore)' TO 'a dashboard' with false positive, new vulnerabilities, vulnerabilities with correction plan, vulnerabilities w/o correction plan)

Section 3: Proposed Methodology on Network Data

3.1. Network Crowdsource Measurement

Determining the quality and extent of coverage is an integral part for any cellular network service provider. It is important to identify the level of coverage, previously Telenor was capturing and analyzing network data and quality of service factors at network level by dealing with network data. But now Telenor has begun to deal with crowdsource network measurement data known as Tutela crowd source data. Through this crowdsource data, it is possible to gain insights beyond the network layer and understand the quality of network performance and experience at the application and end user level [44].

3.1.1. Data Source

Data collected with a global panel of more than 300 million smartphone users. As described in section 2.2. the Tutela collects data and performing tests in the network over a period of 2 years through software embedded in a variety of over 3000 consumer applications and devices. Its providing metadata for 2G, 3G, 4G, 5G signals metrics and categorize them as follows [46].

1. Device data
2. Connection data
3. Location data
4. Quality data
5. Video data

3.1.2. Methodology

From previous literatures on crowdsource data [44][45] for different regions we are highlighting related measurement approaches to devise predictive models for network coverage and quality of experience from end user level by applying machine learning algorithms. Which are capable of capturing complex relationships in the dataset. In a literature [44] they fuse the crowd source data for LTE cellular systems with other information about user's context and radio access network (RAN) configuration to build a predictive model of wireless coverage.

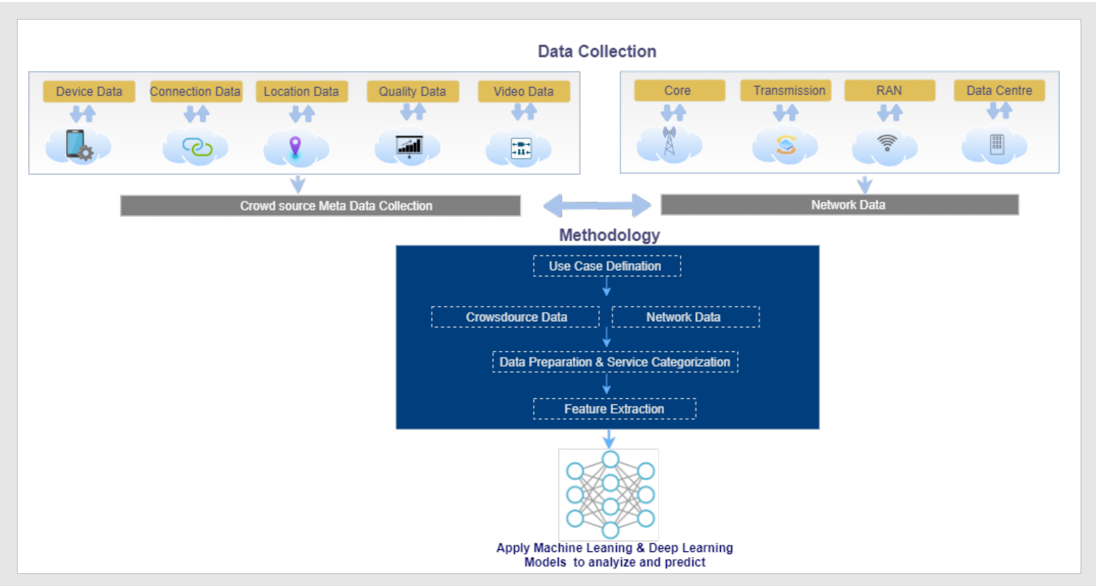

Figure 3. Network Crowdsource Measurement using machine learning

3.1.3. Feature Selection and Engineering

To improve the predictive power the contextual features from the crowdsourced data for example phone model, battery level, user's speed, geo-coordinates and region etc. can be combined with the RAN data available in the networks existing tools for example Subtonomy, Mycom for features like serving cell's location, transmit power, antenna details and frequency band etc. [45]. Feature selection methods that can be used to determine the mobile Internet experience from end user perspective are

1. SHARP
2. Lime
3. X Gradient boosting machine XBG

XGB, Lime [46] and Sharp [44][46] methods are already used by the Telenor to diagnose quality of experience QOE problems for their IPTV services from their network data[46].
From the literatures on crowdsource data and the work already done by the Telenor it is determined that SHAP is most suitable approach for diagnosing the problems, their accuracy on pinpointing the culprit feature and localizing the problem to a specific segment of the network are high [44][46].

3.1.4. Machine learning applications

From the previous literatures, in order to accomplish the task of determining the end-user experience, machine learning ML techniques can be used not only to predict the failure but also to address the specific network segment causing problem [44][46] [47]. The techniques that can be applied on crowd source data are

1. Gradient boosted trees [44][47]
2. LightGBM [44]
3. Random forest [46] [47].
4. XG Boost [46] [47].
5. Multi-layer perceptron MLP [46] [47].

Gradient boosted trees, Random forest and Multi-layer perceptron MLP these methods are already used by the Telenor to diagnose quality of experience QOE problems for their IPTV services from their network data [46] [47]. From the literatures on crowdsource data and the work already done by the Telenor it is determined that the performance of all models depends on the availability of data and particularly the rare events. But the MLP (Neural Networks) performs better as compared to others with the increase in data volume [46] [47].

3.2. Network Performance Control

Understanding and determining the customer experience and satisfaction is a key competitive advantage for telecoms. For this purpose, the operators directly ask their subscribers, through these surveys the subscribers give the NPS score and gives network an idea about their loyalty alternatively this score helps network to determine their churn rate.

This mechanism, although reliable and widely used in the industry, but it is infeasible for real-time or large-scale settings [49]. With these surveys we are not able to measure customer satisfaction at any point in time and identify potential causes of poor customer experience. To cop with this Telenor aimed to work with artificial intelligence and machine learning to identify the relation between the NPS score and their network KPIs to find the effected services, hoping it would be easier for them to address any issue promptly, before they deteriorate and impact a larger number of subscribers. As it lies in the interest of SEMANTIC to be able to improve the network performance by 10 folds as an end target, by estimating the quality of their users' experience. To aid in this one can employ a multitude of data modeling and data mining techniques [45].

3.2.1. Data Source

Telenor is collecting data from their internal data sources as mentioned above including Mycom, Subtonomy etc. they are continuously collecting and monitoring the radio access network performance, capacity and quality. To provide the information about their KPI's for network traffic on cell to user equipment level including calls, data volume and handover attempts [3]. The amount of data which could be gathered from mobile network data sources is vast [48]. From this data Telenor have had more than 90 KPIs available for LTE related technologies. These KPI cover throughput, success rates and utilization amongst other things at various levels of the network [reference for students]. The collected data is categorized as follows

1. NPS data
2. Cell/Site data
3. Connection data
4. Geolocation data
5. Quality data

3.2.2. Methodology

The preparations for the experiments to be conducted in this project are categorized into four sections. [reference to student work]

1. Data gathering
2. Filtering
3. Segmentation
4. Training and testing

In these sections the steps are identified for constructing data sets and the models that can be trained on. It involves creating clusters of cells and selecting those which have received a sufficient number of survey answers to calculate the mean promoter score MPS that is used as a label for training or testing. Key features are selected from the available KPIs which form the time series for prediction. [reference to student work].

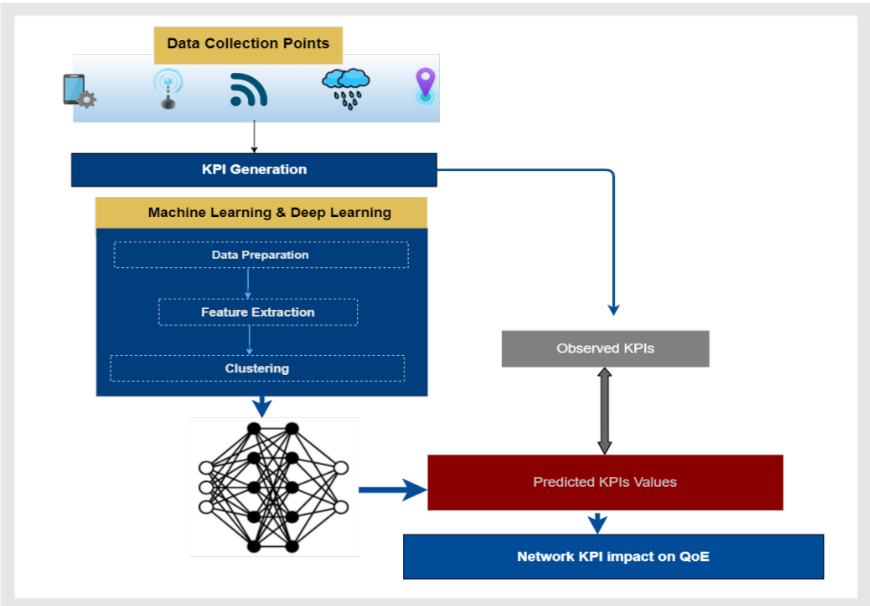

Figure 4. Network Performance Control using machine learning

3.2.3. Feature Selection and Engineering

To improve the predictive power the network data points are grouped together into clusters so that more closely related datapoints are co-clustered to a greater extent than the rest of the data set [50]. There are a number of ways of doing clustering. One method focuses on optimizing the parameters which define the cluster, i.e. position and shape [51]. Clustering methods which are used to group the datapoints together to determine the end user experience against their respective site are [reference to student].

1. K- Means
2. Gaussian Mixture Models

K- Means have the drawback that the shape of each cluster relies heavily on any transformation of the feature space itself, having trouble identifying irregularly shaped clusters on non-linear manifolds. [reference to student]. The other is the density-based clusters which identify a cluster by considering the density of points which are co-located [52]. These methods often hold the benefit of consider local features of the data point distribution which allow them to capture even irregularly shaped clusters as well as identifying outliers.

3.2.4. Machine learning applications

On the basis of the data exploration the correlation is observed between the data feed and the geographical location which helps in predicting user experience. There are number of attributes identified and different machine learning methods are used to find correlations. Important features are identified and then domain expert knowledge is also used to validate the identified attributes. Telenor's work to identify the network KPIs having an impact on the NPS score using deep learning methods. The methods for used for classification are [reference to student work]

1. Artificial neural network (ANN)
2. Recurrent neural network (RNN)
3. Deep, Shallow neural networks
4. Multi-layer perceptron (MLP)
5. Time series analysis

The results help to group the customers experiencing the same network quality on basis of their NPS score.

3.3. Network Disturbance control

Telecom networks are continuously growing and increasing in complexity. With the advancements, the automated services are gaining attention because automations help to improve the service performance and minimizes the downtime. By keeping services up, we can have the potential to achieve the customer satisfaction, the long-term relationship and significant decrease in OPEX. The service automation trend is getting facilitated by Artificial Intelligence and machine learning data driven techniques for proactive anomaly detection and its management before it manifests in the system and results in service downtime or revenue loss for the network operators.

To improve the customer experience, we are considering a case for network control and its automation by applying AI data-enabled mechanisms. Aim is to provide a solution to a "Why did this happen?" question using artificial intelligence to achieve this goal Telenor started working on a problem that is predicting the resolution time required for the network anomaly to fix. This will help to optimize our network by making different key decisions. Backing these decisions with predictive analytics based on historical data analysis our proposed technique includes data collection, preparation and machine learning techniques to quickly understand and detect patterns in machine behavior.

3.3.1. Data Source

Telenor is collecting data from their internal data sources as mentioned above including Splunk, Juniper probes, Remedy etc. These are continuously collecting and monitoring the network performance, its health, and degradations. The data that is used consist of already solved tickets (closed), which are registered in the historical repository of the trouble ticketing system. The information provided by the repository for each ticket refers to the moment when it was closed, as well as all the different actions performed from its creation to its closing. These actions refer to the creation of the ticket, its association with another ticket (so that the solution of one of them induces the solution of the other), its closing, etc. The aim of this project is predicting the resolution time required for the ticket generated by the system. For this purpose, different parameters which are responsible during the life cycle of a ticket are considered and the collected data is categorized as follows [reference to student work]

1. Environmental data
2. Logs interaction Data
3. Ticket creation data
4. Geolocation data
5. Technological data

3.3.2. Methodology

The preparations for the experiments to be conducted in this project are categorized into following sections. [reference to student work]

1. Data gathering
2. Data preprocessing
3. Training and testing

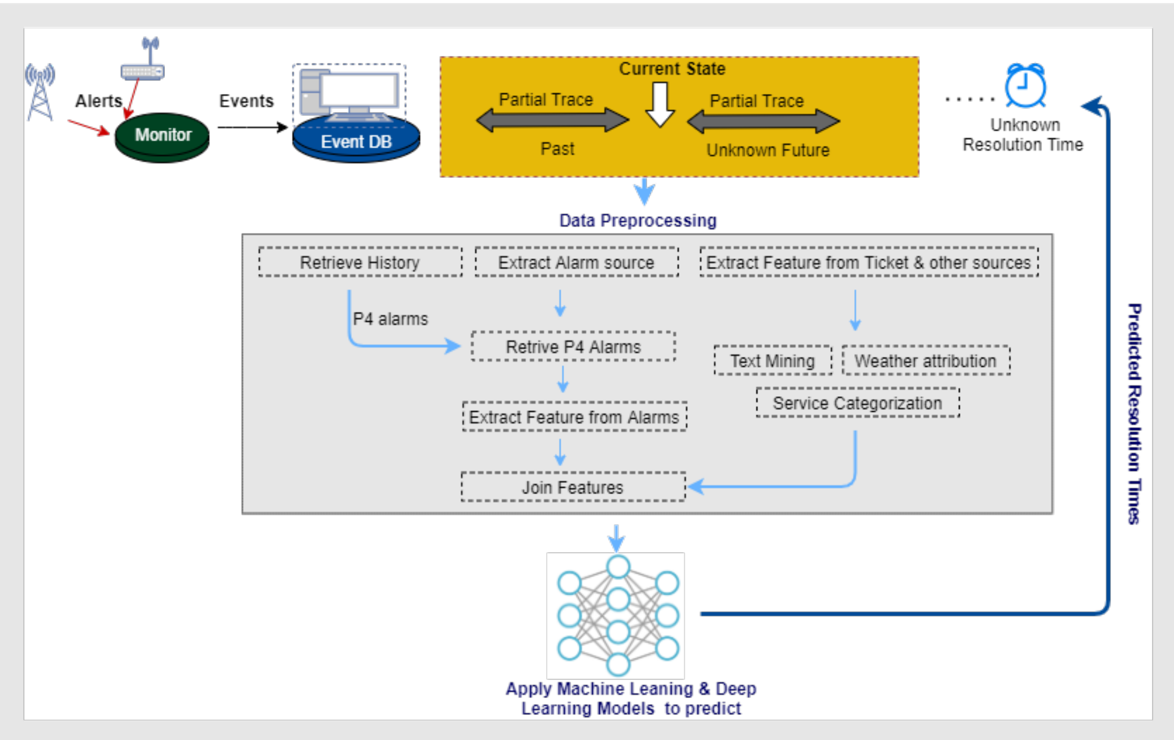

Figure 5. Trouble ticket resolution Time prediction using machine learning

In these sections the steps are identified for constructing data sets and the models that can be trained on. It involves data collection from different sources the challenging part is the trouble ticket data contains unstructured data as well. Different data preprocessing techniques are applied to restructure the data. Moreover, the information obtained against each ticket do not always contain complete or relevant information for all the required attributes.

3.3.3. Feature Selection and Engineering

The data collected from different sources varied in its structure and the information it contains. The information obtained against each ticket do not always contain complete or relevant information for all the required attributes. Besides, some of the attributes are filled in with unstructured text, which is not directly manageable by machine learning techniques. They are carefully analyzed in order to extract as much information as possible. Some text mining techniques have been used, such as [53] [reference to student work].

1. Stemmer algorithms
2. Frequency recounts
3. Term based frequency
4. RegEx function

These feature engineering techniques helps to restructure the unstructured data and extract important features from them. RegEx is a regular expression-based technique that identifies the specific patterns and work by 'find' or 'find and replace' operation, thus it breaks the unstructured data, but it requires expertise. While other data mining techniques involves number of steps and time consuming [53] but they help to identify the important and most recurrent features on basis of the term-based frequency.

3.3.4. Machine learning applications

There are huge number of attributes it has been necessary to perform an attribute selection to keep the most relevant ones. For this purpose, different machine learning and deep learning algorithms are applied that are more suitable for the solution of these particular problems. Selected algorithms are applied for feature selection then domain expert knowledge is also used to validate the identified attributes. Telenor's work to predict trouble ticket resolution time evaluated several classification and regression models [54] [reference to student work] Classification and regression methods used for prediction are [reference to student work]

1. Gradient boosting machine
2. X-Gradient boosting machine
3. Decision trees
4. Random forest
5. Neural networks
6. K means clustering

The results help to partition tickets into different time bins. The successful classification of tickets into these categories provides a sufficient granularity about their resolution time [reference to student work].This improvement suggests the possibility of deploying a model to the network that proactively estimates the network outage duration and thus help organization to take mandatory actions to control the network.

3.4. Service Quality Prediction

IPTV are getting popularity in television content over the network then traditional channel delivery methods which results in production of largest volume on IP networks. Its challenging to minimize service disruptions as even small disturbances in video streaming can quickly result in the user impatience and dissatisfaction. Telenor aims to improve Quality of Experience (QoE) while keeping the costs margins lower and improving the service monitoring capabilities so they can act proactively and mitigate any service degradation. Telenor worked with Ericson's to design an effective root cause inference sub-system which can be implemented as submodule in real time service assurance system for improving IPTV QoE.[46][47]. To accomplish this task Telenor worked on their home Wi-Fi data by applying AI/ML AI/ML techniques are used to not only predict the failure but more importantly pinpointing the specific network segment causing problem [46][47]. The improvement in QoE for IPTV by automations lead to

1. Downtime minimization
2. Performance improvement
3. Opex minimization

3.4.1. Data Source

Data collected for the IPTV analysis is collected over the period of 95 days spanning a time period from end of 2018 to start of 2019 as described in section 2.3. the STB has a proprietary probe that reports objective measurements of QoE every 5 minutes. In addition to QoE measurements, Wi-Fi radio data is collected from each household other than household and QoE measurement data is collected from packet switches and routers as well and the collected data is categorized in to three different sources namely [46][47].

1. STBs dataset
2. Wi-Fi Radio dataset
3. Links Traffic throughput dataset

3.4.2. Methodology

In these sections the steps are identified for constructing data sets and the models that can be trained on. It involves data collection from different sources as described in section 3.4.1. The challenging part is to deal with missing values. Different data preprocessing techniques are applied, and data categorized into four main classes [46][47]. There are four Classes:

1. OK
2. Major
3. Minor
4. Unavailable.

These classes are defined on basis of their QoE value. The QoE is measured only when STBs are active and to determine their class (major, minor) their activity levels are analyzed.In the inactivity periods, STBs report zero values for all QoE classes, which allows to track the activity level for different users for analysis [46][47]. The activity levels for each user vary significantly across the households. Another property of this data is its very unbalanced where most users do not experience any QoE problems most of the time so data samples where they do experience QoE problems is significantly less, which is challenging for learning high quality ML models [46][47].

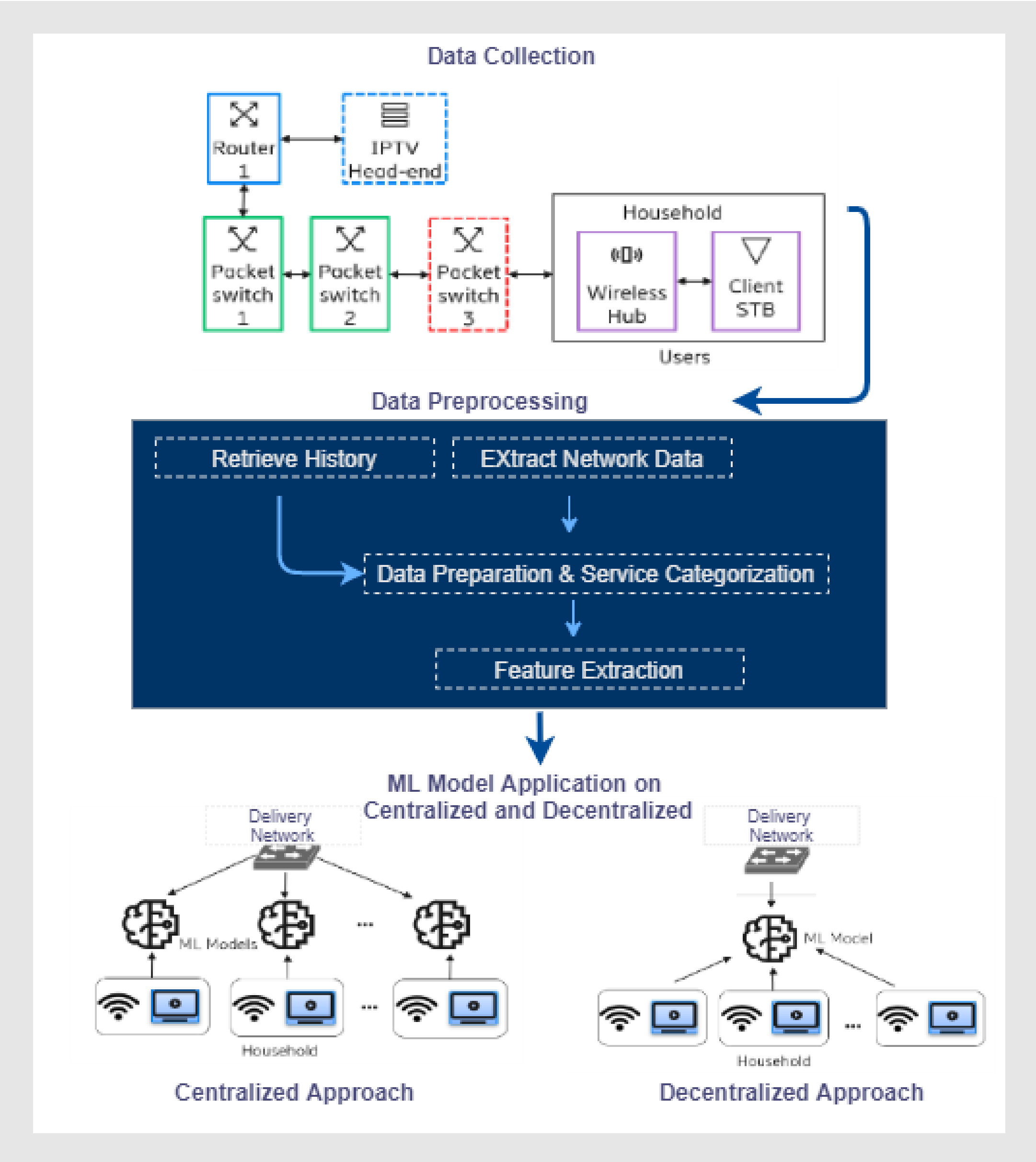

Figure 6. Network Service Quality Prediction using machine learning

3.4.3. Feature Selection and Engineering

73 features are extracted from these data sources to describe a Wi-Fi connection state. Its challenging to work with home Wi-Fi data because of its missing values. Missing data is catered by setting a rule 30sec < t Missing data > 5mins where small intervals less than 30 secs are skipped, and the large intervals are filled with mean. Lime and Sharp [46][47] methods are used by the Telenor to diagnose quality of experience QOE problems for their IPTV services from their network data. [46].

1. SHARP
2. Lime

Few of the important features identified and used for training the models as described in [46][47] are

1. Available bandwidth
2. Physical rate
3. Channel number
4. Broadcasts and discarded packets
5. Multicast, Unicast and Error packets
6. Received and transmitted kilobytes
7. Wi-Fi frequency band

From the work done by the Telenor it is determined that SHAP is most suitable approach for diagnosing the problems, their accuracy on pinpointing the culprit feature and localizing the problem to a specific segment of the network are high [46][47].

3.4.4. Machine learning applications

The approaches used for IPTV QoE has been active area of research and has been explored quite extensively in the past including conventional machine learning methods for centralized and decentralized. And the performance metrics used includes F1 score and Recall [46][47]. The models used for analyzing the performances uses 70% of data for training that holds data for 67 days and 30% for testing from last 28 days. The ML models used are [46][47].

1. Random forest
2. XGradient Boosting machine XGB
3. MultiLayer Perceptron MLP

As Telenor aimed not only to predict the failure but more importantly pinpointing the specific network segment causing problem for this purpose the model interpretability or explain ability techniques are classified into two groups as per [46][47].

1. Model-based (Whitebox)
2. Model agnostic techniques (Blackbox)

It is analyzed that centralized models are better in capturing and predicting the QoE problems. But decentralized are useful in scenario where there is a need of monitoring and managing particular node only if target node has the availability of a big and balanced dataset.
The performance of all models depends on the availability of data and particularly the rare events. MLP (Neural Networks) performs better as compared to others with the increase in data volume [46][47].but the training time is much longer for the MLP as compared to the RF and XGB models. So, there is tradeoff between time and accuracy all these models can perform will depending on the scenarios.

3.5. AI/ML innovation to O-RAN

At Nokia, we take the help of the Nokia's SEP (Service Enablement Platform) [76] in order to build xApps leveraging AI/ML upon it. SEP is the name of the Nokia product that implements the O-RAN defined near real-time RAN Intelligent Controller (RIC) functionality, Multiaccess Edge Computing platform (MEC) functionality and related Nokia specific added value functions (Refer to Figure 3). Edge Cloud datacenters and Multi-Access Edge Computing (MEC) are about user experience and new revenue opportunities. User experience is primarily driven by higher bandwidth and lower latency use cases, whereas revenue opportunities are driven by a diverse set of new 5G use cases at the Edge. Open RAN (O-RAN) alliance is about defining an open, decomposed RAN that is fully programmable. Key element in this programmability is the RAN Intelligent Controller (RIC), translating high level operator defined service level policies into actionable RAN resource related methods so that the use of radio resources is optimized and end user perceived Quality of Experience (QoE) is maximized for any given dynamic condition in accordance with said policies. The SEP is all about combining these two, the MEC and RIC together to provide unique value by connecting the service awareness of the MEC and the RAN control of the RIC to enable service-aware RAN optimization.

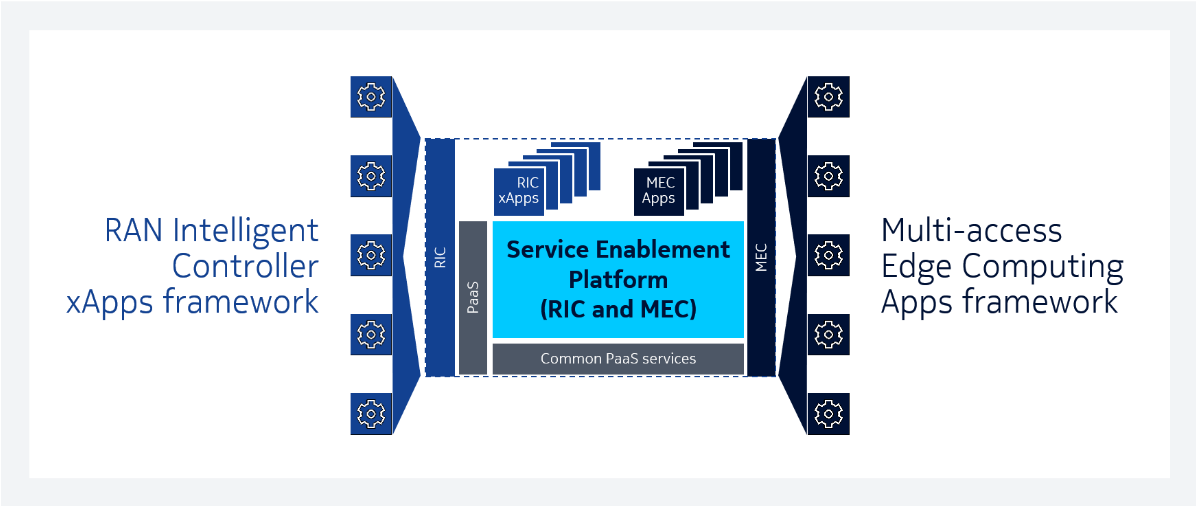

Figure 7. SEP encapsulating RIC and MEC [76]

The high-level view of placement of functional RIC platform in SEP is shown in figure 4:

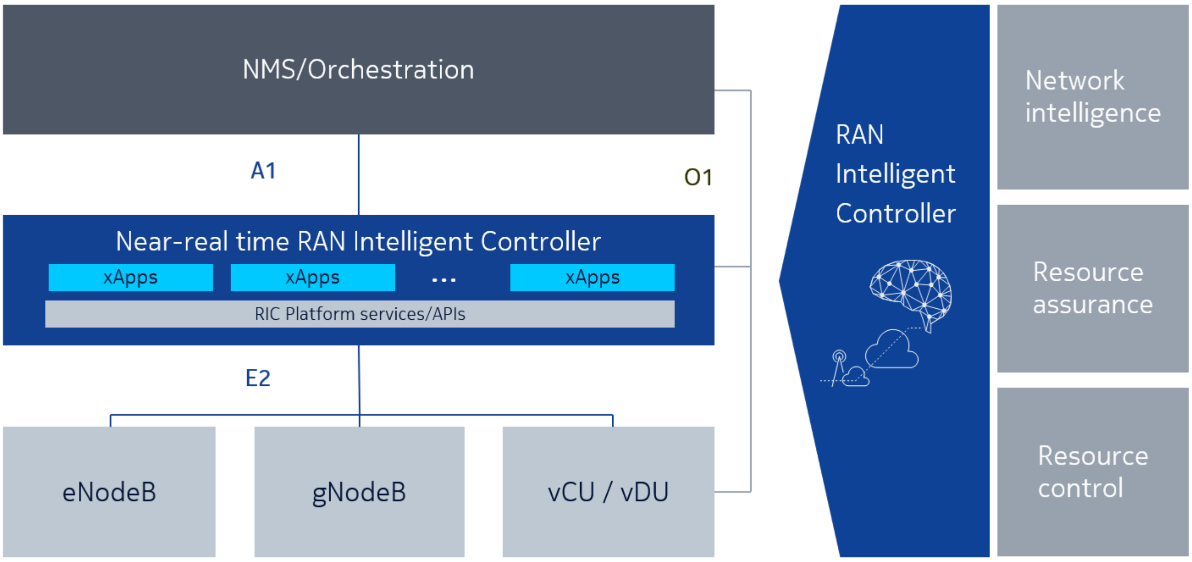

Figure 8. Overview of the RIC functional architecture with disaggregated RAN nodes [76]

Future cloud-native mobile networks will rely on edge-cloud based architecture including new devices, massive scale radio access and a plurality of connected converged cores. They will be programmable (e.g. SDN, NFV, cloud-native) and will be augmented via cognition brought with the help of analytics and machine learning layers. A deep literature review and POC of various use cases around ML lifecycle and xApp influencing the functioning of RIC suggests provisioning of new-services in sub-minute scale. This helps in realizing the SEMANTIC KPI set for Nokia.

3.6. AI/ML in network control and management

Software defined networking (SDN), network function virtualization (NFV) and end to end network slicing (E2ENS) empower 5G networks and beyond to satisfy various demands and services including but not limited to ultra reliable and low-latency communication (URLLC) enhanced mobile broadband (EMBB) and massive machine-type communications (MMTC). 5G networks utilize SDN, VNF and E2ENS for dynamic and efficient resource allocation and network management.

Network slicing can be performed at different parts of the network, including core cloud, edge cloud, radio resource management, radio access network (RAN) processing, spectrum, and radio frontend to manage and allocate different resources to satisfy the different service demands [111].

A virtualized network slice consists of a number of Virtual Network Functions (VNFs) distributed geographically in numerous Data Centers (DCs). Each VNF provides certain services in its slice and all the VNFs of a slice collectively provide wireless network access to the User Equipments (UEs) attached to that slice [112].

Leveraging network function virtualization, network slicing enables multiple virtual networks, i.e., network slices, run on top of a common physical network infrastructure. Each network slice can be tailored to meet the diverse network requirements of a specific use case. In network slicing, slice tenants have different service level agreements (SLAs) with the mobile network operator, e.g., slice throughput and end-to-end latency, and have full control of the operation of their slices, e.g., resource management and user admission control. The objective of network slicing for the network operator is to efficiently utilize network resources to maximize the overall network utility such as throughput, latency, and revenue and meet the SLAs of slices, which boils down to a network utility maximization problem. In the literature, the network utility maximization (NUM) has been extensively studied. In these works, NUM is usually formulated as an optimization problem with given mathematical models and solved by various optimization methods, e.g., gradient descent methods. However, the mathematical expression of utility functions of users can be very complicated and difficult to obtain in real network circumstances. On one hand, the utility functions of users are affected by multiple factors, e.g., channel condition, user traffic, and network workload. It is hard to obtain the closed form mathematical models especially in highly dynamic mobile networks. On the other hand, slice tenants have their own customized slice operation strategies, e.g, user admission and scheduling. These control strategies, which can be time varying, change the utility of network slices. As a result, it is impractical to assume the closed-form expression of utility functions in optimizing the resource allocation in network Slicing [113].

The resource allocation in network slicing is a highly complicated problem, which the existing traditional approaches can not solve effectively and efficiently. First, traditional optimization approaches require accurate mathematical models with parameters known, which is often difficult to achieve in practice, especially with the increasing complexity, scale and service diversity of the 5G and future networks. Constraints from the physical systems and service demands are prevalent and complex, such as latency requirement, service level agreement (SLA) and safety demand, which further adds to the difficulty, let alone obtaining a closed-form expression. Second, traditional methods do not adapt to epistemic uncertainty, exhibited as hidden structures in networks, due to a lack of knowledge and subsequent ability to explore and learn from the studied system. Faced with these challenges, learning-based approaches are beneficial because they explore and learn from the environment without assuming the knowledge of accurate models. Recently, there have been growing learning-based network research works showing significant performance improvement [111].

5G networks are expected to be able to support a large number of tenants simultaneously with different Quality of Service (QoS) requirements and also with a wide range of services with different service level agreements (SLA). Satisfying these expectations makes optimal resource allocation a crucial key for the success of 5G networks. In fact, optimal resource management is a significant challenge in the context of 5G networks for the researchers. It is impractical to use traditional optimization approaches therefore in order to meet the QoS requirements for each network slice and satisfy SLA for each tenant Artificial Intelligence (AI) approaches are getting more and more attention. Utilizing AI for Automated resource management is one of the main state-of-the-art research topics in 5G networks and beyond. There are many challenges in efficient implementation of E2ENS and VNF and also in optimal resource allocation. To this end, advanced Machine Learning (ML) algorithms (Deep Learning (DL) and Reinforcement Learning (RL) algorithms) will be utilized for processing the big data volume in different parts of the network (MEC, RAN and core), in order to automate network control and dynamically allocate heterogeneous resources (computation, storage and bandwidth) in an end-to-end (E2E) method over different domains of 5G networks.

3.6.1. Machine learning methods for resource allocation

In network slicing, dynamic resource orchestration and network slice management are critical for resource efficiency. However, it is highly complicated such that the traditional approaches can not effectively perform resource orchestration due to the lack of accurate models and hidden problem structures [111]. To address this challenge, the authors in [111] propose a constrained reinforcement learning based approach for network slicing. The resource allocation problem is modeled as a Constrained Markov Decision Process (CMDP) and is solved using constrained reinforcement learning algorithms for network slicing under both cumulative and instantaneous constraints. Specifically, to deal with cumulative constraints, an adaptive constrained reinforcement learning algorithm based on Interior-point Policy Optimization (IPO) is proposed and for instantaneous constraints, the resource allocation decision generated by reinforcement learning algorithm is projected to its nearest feasible decision at the end of policy neural network [111].

In [114], a new dynamic Edge/Fog Network Slicing scheme (EFNS) is presented in which the Infrastructure Provider (InP) allocates edge/fog resources (communication, computation and storage resources) to slices on demand and serves requests from multiple tenants for some reward. The scheme allows the InP to make additional profits by periodically updating the allocation of resources to slices based on the demand and resource utilization. Furthermore, when the InP cannot accommodate the demand with its existing resources in stock, it can buy back surplus resources from tenants. The dynamic variations in subscribers' traffic results in available resources that tenants can lease back temporarily to the InP. To efficiently use these resources and augment the infrastructure of the InP, tenants can also lease their subscribers' mobile edge devices whenever idle. The available resources combined with the edge devices' capabilities create fog nodes that can act as access points and relay traffic for other users. Therefore, edge devices, when acting as fog nodes, can provide connectivity for other users, temporarily augmenting the InP infrastructure. Slicing these resources and reselling them to other tenants increases the revenue of the InP while tenants recover part of their investment, resulting in a win-win situation. A semi-Markov decision process is used to model the arrival of slice requests by taking into account the dynamics of users' demands and availability of resources. To find the optimal slice request admission policy, which includes slices with augmented resources, in order to maximize the long-term revenue of the InP under uncertain resource demands, a Q-learning (Q-EFNS) algorithm is developed. Additionally, to improve the convergence time and reduce the computational complexity of Q-learning in large-scale scenarios, a Deep reinforcement learning (DQ-EFNS) algorithm and an enhancement based on a Deep Dueling (Dueling DQ-EFNS) algorithm are presented for dynamic edge/fog network slicing [114].

With today's networks becoming increasingly dynamic, heterogeneous, and large-dimensioned, the real-time tracking and explicitly modelling of the networking environment are getting increasingly costly or even intractable. It is strongly vulnerable to environmental changes for the slicing schemes that customize the immediate performance over a given deterministic environment information in the literature. This requires that a slicing system should be able to make decisions in the absence of partial system state information while the resulting solution efficiencies are safeguarded across the whole running trajectory [115]. In [115] the authors first present a two-stage slicing optimization model with time-averaged metrics to safeguard the network slicing in the dynamical networks, where prior environmental knowledge is absent but can be partially observed at runtime. Directly solving an off-line solution to this problem is intractable since the future system realizations are unknown before decisions. To solve this issue a learning augmented optimization approach with deep learning and Lyapunov stability theories is proposed. This enables the system to learn a safe slicing solution from both historical records and run-time observations [115].

Network slicing enables operators to isolate and customize network resources on a per-service basis. A key input for provisioning resources to slices is real-time information about the traffic demands generated by individual services. Acquiring such knowledge is however challenging [116]. In [116], a framework to demand estimation for sliced network Management and Orchestration (MANO) is presented. The target is the inference of service-level demands that are required for capacity provisioning to individual slices. The proposed approach hinges on the concept of mobile traffic decomposition (MTD), (the process of breaking down aggregate traffic time series into individual service demands). The proposed framework solves the MTD problem effectively by feeding suitably transformed mobile network traffic to interchangeable deep neural network architectures, including a new class of 3D Deformable Convolutional Neural Networks (3D-DefCNNs) that is explicitly designed for decomposition [116].

In [117] a Deep Neural Network (DNN) architecture based on a Convolutional Neural Network (CNN) is proposed that uses the traffic measurement data of a slice to make a capacity prediction for that slice. The architecture forecasts the capacity needed to accommodate future traffic demands within individual network slices while accounting for the operator's desired balance between resource overprovisioning (allocating resources exceeding the demand) and service request violations (allocating less resources than required). The loss function is defined in order to assure that no underprovisioning will occur and overprovisioning will be minimized. This loss function makes the architecture stand out from other forecasting algorithms that predict based on minimization of the total error (both underprovisioning and overprovisioning) of the actual and forecasted future capacity [117].

Forecasting future resource demands of network slices utilizing machine learning algorithms (mainly deep learning and reinforcement learning) has become one of the main tools for resource management in 5G networks and beyond. The output of these algorithms can be used to design policies for network control and allocation of resources. One critical challenge is designing an appropriate algorithm where high speed and accuracy as well as low computational usage of the algorithm are the key features for selecting an algorithm. A lot of the research works in the current state of the art has been done and is being done in order to improve the accuracy of the predictor, which by itself can not guarantee that the final designed framework for resource management is optimal. One enhancement could be to extend the current state of the art by developing more refined forecasting algorithms that are trained using information relevant to 5G networks and beyond. Designing frameworks based on these novel algorithms could lead to better optimal policies for network control and resource allocation. Another enhancement which is gaining attention nowadays is to design distributed and decentralized algorithms for predicting the future demands which will lead to distributed resource allocation and network control frameworks.

3.6.2. Distributed machine learning approaches

5G networks are expected to reach an enormous scale and high speed and ultra reliable communication services. These expectations lead the research on slice management to gain attention on distributed methods and algorithms.

In contrast with centralized solutions which are commonly based on Software Defined Networking (SDN), distributed schemes are implemented by distributing the actors of the algorithms among the data centers and slice providers and also define the actions and operations such that the result of the actions of all the entities provide optimal resource allocation to the slice Virtual Network Functions (VNFs) [112].

Exploiting deep learning and deep reinforcement learning (DRL) for resource management in mobile networks has gained increasing research attention. These works formulate the network utility maximization problem as a reinforcement learning problem and apply DRL techniques such as Deep-Q Learning to solve the problem. It is shown that DRL obtains considerable improvement on the system performance in terms of throughput, latency, and utility. However, these solutions are centralized network resource management which does not allow individual network slices to manage their own resources. Moreover, these solutions are designed for solving unconstrained optimization problems. As a result, they cannot guarantee the service level agreements (SLAs) of network slices in resource allocation [113]. To effectively allocate network resources to slices, the authors in [113] propose a framework that integrates the alternating direction method of multipliers (ADMM) and deep reinforcement learning (DRL). The algorithm decomposes the network slicing problem into a master problem and several slave problems. The master problem is solved based on convex optimization and the slave problems are handled by the corresponding network slices so that the isolation among slice tenants can be ensured. Since there is no closed-form expression of the utility functions of users in slave problems, the Deep Deterministic Policy Gradient (DDPG), which is a state-of the-art DRL technique, is used to learn the optimal policy and allocate the resource to users accordingly [113].

In [118], a cross-domain resource orchestration solution for dynamic network slicing in cellular edge computing is studied. The fundamental research challenge is from the difficulty in modeling the relationship between the slice performance and resources from multiple technical domains across the network with many base stations and distributed edge servers. To address this challenge, a distributed cross-domain resource orchestration protocol which optimizes the crossdomain resource orchestration while providing the performance and functional isolations among network slices is proposed. The main component of the protocol is a distributed cross-domain resource orchestration algorithm which is designed by integrating the alternating direction method of multipliers (ADMM) method and a new learning-assisted optimization (LAO) approach. The edge node is defined as a logic network unit consisting of a physical base station and a certain amount of computing resources, e.g., an edge server. Then, cellular edge computing can be recognized as a collection of interconnected edge nodes. ADMM is applied to decompose the cross-domain resource orchestration into two subproblems. The first subproblem handles the resource orchestrations within an edge node while the second subproblem coordinates the resource allocation among edge nodes. Since the slice performance model is not available, the performance of a slice is presented using a black-box function. Therefore, the first subproblem becomes a black-box optimization problem which is solved by designing a new learning-assisted optimization algorithm that constructs a probabilistic model for the black-box function and iteratively learns the gradients of the function with the observed data for the optimization. The second subproblem is a standard quadratic programming problem, and is solved based on convex optimization [118].

In [112], a resource allocation model for 5G virtualized networks in a heterogeneous cloud infrastructure is proposed. In this model, each network slice has a resource demand vector for each of its virtual network functions. First for a system of collaborative slices the resource allocation formulated as a convex optimization problem, maximizing the overall system utility function. Then a distributed solution for the resource allocation problem by forming a resource auction between the slices and the data centers is introduced. For a system with non-collaborative slices, a resource allocation problem based on the notion of dominant resource fairness is formulated and a fully distributed scheme for solving the problem is proposed [112].

Another distributed method that has become a very important research topic is Federated Learning [119]. This approach is a recent machine learning technique that allows multiple decentralized agents to train a shared model holding their local data without exchanging them. Federated Learning contrasts with traditional centralized machine learning where all datasets are aggregated in one central server and also it contrasts with some decentralized methods in which the local data samples have identical distribution. In particular, Federated Learning puts together elements from large-scale machine learning, privacy preservation and decentralized optimization.

Conclusion

This document has summarized the requirements for Inter Slice management and joint allocation of resources in MEC/RAN clouds. It includes the identification of state-of-the-art tools, methodologies and approaches for network control and its automation. We started with an 'Introduction' of the automation methodology of telecom networks, detailing how identification of potential network KPIs that affect the end user experience will provide a control and help to make our network automated and intelligent enough to translate the real-time analytics into actionable insights and capable of re allocating resources smartly.

Following that in 'State-of-the-art tools', we then discussed relevant tools that will allow us to improve network performance by gaining control over it. To address this, we described the experience at Telenor, in which we developed different use cases and then tested these hypotheses by collecting data from different network domains. As a result of their successful experiments, Telenor proposed various tools that are already implemented and provide data across different network domains. The information collected from different network domains includes RAN, Transmission, Core, Fixed network Wi-Fi, Cloud, to state some of the most relevant domains.

These identified network tools will finally enable the methods that are projected to be deployed in the mid- and long-term of SEMANTIC project while maintaining and improving the network performance. Many of the future requirements for network control and its automation can be supported and derived by the 'Proposed Methodologies', as emphasized in this part of the document. In this Section discussion of potential methods elucidated how specified open research questions can be tackled by applying AI- enabled mechanisms.

Again, based on the work done by Telenor, different AI-based models served as a base to the answer the questions for example ''what happened? ''why did this happen''? ''when it can get fixed?'' The explored models referenced to Section 3 with their detailed methodologies will help to optimize our network by making different key decisions. Backing these decisions with predictive analytics based on historical data analysis our proposed techniques include data collection, preparation, and machine learning models to quickly understand and detect patterns in machine behavior. From the approaches applied it is identified how centralized models are better in capturing and predicting the QoE problems and decentralized useful in scenario where there is a need of monitoring and managing particular node only if target node has the availability of a big and balanced dataset.

The work on trouble tickets resolution time modelling referenced to in Section 3, suggests the possibility of deploying a model that proactively estimates the network outage duration and thus help organization to take mandatory actions to control the network. Various experiments on different network domains provide better picture of mobile network performance from a specific user group, in different regions of the country against various network technologies. This targeted identification of services and their performance can help improving the capability to re-allocate the resources as per the size of the pooled resources and the time required. Ultimately it will help to achieve the SEMANTIC KPI and target value of 10-fold improvement of the capability to re-allocate resources compared to today's capabilities.

To sum up, while ensuring that the identified network tools meets the demands of current and near future projects, the work toward Semantic Project will continue: development of algorithms, processes and models for automated, data-driven network resource (network slices) control and policy-based network utilization optimization.

ANNEX I. Other tools explored during the SoA review

In addition to the tools used by Telenor, some other tools have been explored that can be used across different network domains for configuration, monitoring, data collection, analysis troubleshooting are in place. Some relevant examples follow below:

1.1. RAN Domain tools

- **Infosphere Data stage by IBM**
  It is an end-to-end data integration tool that provides a complete set of capabilities to understand, cleanse, monitor, transform and deliver data. Its capable to provide rapid prototyping, profiling, automated data validation, testing advanced data transformation techniques and connectivity to Cloud Applications. [20]

- **Azure Stream Analytics by Microsoft**
  It is an event processing engine that enables development and real-time analytics on multiple streams of data from sources such as devices, sensors, web sites, and other applications. [21]

- **Cloudify**
  It is an open source, end-to-end platform designed to transform network services and multi-cloud applications to manage different automation as part of one common CI/CD pipeline. [22]

- **NETSCOUT**
  It provides a complete, real time view of the radio access network for service assurance, small cell planning and RAN optimization. [23]

- **IBM Workload Automation**
  It is used for batch and real-time workload management, available for distributed mainframe or hosted in the cloud. It helps to drive workloads on hosted servers, at low costs for central server. [24]

- **Logic monitor**
  It cloud-based monitoring tool supports SD-WAN and cloud-based networks. It helps in automated setup and configuration with dynamic discovery of new network devices. It provides secure monitoring for firewalls, routers, switches, wireless devices, load balancers and more. It also provides network topology mapping, root cause analysis, and intelligent alerting. [25]

- **Amazon Cloud Watch**
  It provides infrastructure and application monitoring and its management. [26]

- **Alibaba ChaosBlade**
  It is an open-source troubleshooting tool to inject experiments in the systems to tests the failures. [27]

- **Dynatrace software**
  Its all-in-one, AI powered, fully automated infrastructure monitoring software Intelligence platform that provides monitoring for application performance, digital experience, business analytics and AIOps. [28]

1.2. Transport Domain tools

- **Beam Planner**
  It is a cloud-based planning and optimization software that uses artificial intelligence and machine learning techniques to predict traffic patterns and can help in optimizing the wireless capacity and coverage. [29]

- **Nova by EXFO**
  It's provides a platform that automatically detects, predicts, and diagnoses customer impacting events in complex and dynamic networks. It helps to detect and diagnose outages and degradations by making end-to-end link testing and proactive monitoring in fiber networks. [30]

1.3. Core Domain tools

- **Chef**
  It is a configuration management and cloud infrastructure automation tool that turns infrastructure into code and allows to automate build, deploy, and manage infrastructure. [31]

- **Section**
  It is an edge compute platform to provide an access to real-time metrics thus enables increased visibility across development, testing, acceptance and production. [32] [33]

- **Terraform**
  It is an open-source infrastructure as code software tool created by HashiCorp. It allows users to define and provision data center infrastructure to reduce errors and simplify recovery. [34]

- **Amazon Cloud Watch**
  It provides infrastructure and application monitoring and its management. [35]

- **Alibaba ChaosBlade**
  It is an open-source troubleshooting tool to inject experiments in the systems to tests the failures. [36]

1.4. Edge cloud Domain tools

- **Infosphere Data stage by IBM**
  It is an end-to-end data integration tool that provides a complete set of capabilities to understand, cleanse, monitor, transform and deliver data. Its capable to provide rapid prototyping, profiling, automated data validation, testing advanced data transformation techniques and connectivity to Cloud Applications. [37]

- **Azure Stream Analytics by Microsoft**
  It is an event processing engine that enables development and real-time analytics on multiple streams of data from sources such as devices, sensors, web sites, and other applications. [21]

- **Section**

It is an edge compute platform entirely focused on DevOps. It allows developers to deploy a workload where, when and how they need to by providing them an access to real-time metrics. [33] [32]

- **ChartMogul**
  It is an analytical tool that provides data ingestion, analysis, insights, and customizability. It is well known for its geographical heat map feature to determine where revenue and customers are coming from. [38]

- **ONCITE by German Edge Cloud**
  It is an industrial edge cloud service that provides insights into the production data through real-time data availability and supplementary network artificial intelligence (AI) capabilities. [39]

- **AWS CloudFormation**
  It resource management platform that provides the resource provisioning and updating in a predictable way. [43]

- **Azure Automation**
  It can be used to automate and manage tasks and to orchestrate actions across external systems from right within Azure. [40]

- **Cloudify**
  It is an open source, end-to-end platform designed to transform network services and multi-cloud applications to manage different automation as part of one common CI/CD pipeline [22]

- **Istio**
  It is a completely open source service mesh that is used for observing the services. Istio's diverse feature set provides a uniform way to secure, connect, and monitor microservices. [42][41]

- **Amazon Cloud Watch**
  It provides infrastructure and application monitoring and its management. [35]

1.5. Cloud and their candidate technologies

Name	Cloud	Training	Inference	Maturity	Model-Based	OpenSource	Big Data support
AWS [77]	Public	Yes	Yes	High	Yes	No	No
Azure [78]	Public	Yes	Yes	High	Yes	No	Yes
K2 [79]	Ready	No	Yes	Low	Yes	No	No
AVA [80]	Private	Yes	No	Med	No	No	Yes
MLFlow [81]	Ready	Yes	Yes	Med	Yes	Yes	No
KubeFlow [82]	Ready	Yes	Yes	High	Yes	Yes	No
Valohai [83]	Ready	Yes	No	Med	Yes	No	No

Hortonworks [84]	Ready	Yes	No	High	No	Yes	Yes
Cloudera [85]	Ready	Yes	No	High	No	No	Yes
OpenVino [86]	Ready	No	Yes	Low	Yes	No	No
TensorRT [87]	Ready	No	Yes	Med	Yes	Yes	No
TFX [88]	Ready	Yes	Yes	Med	Yes	Yes	No
CDLK [89]	NA	No	No	Low	No	Yes	Yes
Acumos [90]	Ready	Yes	Yes	Med	Yes	Yes	No

1.6. List of candidate ML frameworks

Name	Language	Usage	Neural Net GPU Support
scikit-learn [91]	Python	High	No
PyTorch [92]	Python, C++	Med-High	Yes
Tensorflow [93]	Python (Keras), C/C++, Java, Go, JavaScript, R, Julia, Swift	High	Yes
Caffe [94]	Python, MATLAB, C++	Med	Yes
Theano [95]	Python	Med	Yes
Spark (MLLib) [96]	Java, Scala	High	No
Deeplearning4j [97]	Java, Scala, Clojure, Python, Kotlin	Med	Yes

1.7. List of candidate pipeline engines

Name	Description
Luigi [99]	- Spotify made Python based workflow mgmt - Workflows defined in Python
Airflow [99]	- Python based workflow - Includesthe server responsible for orchestration - Shell and Python operation suppor
Argo [100]	- A container executor for Kubernetes - DAG system

	- Retries, volumes, parallelism, conditionals, etc.
Brigade [101]	- An event driven FW for Kubernetes
Fission [102]	- Fission is an open source,Kubernetes-native Serverless functions framework with support for public, private, and hybrid clouds
Spinnaker [103]	- Tuned for the application delivery
Kubeflow [104]	- ML tailored workflow for K8s including, Argo WF engine, JupyterHub for development, Pachydermand SeldonIO for management, Tensor2Tensor for model mgmt., TFX libraries for serving (inference) transform (pre-processing) and model analysis (evaluation)
Doit [105]	- Pure python, build tool kinda thingy - Workflows defined in Python
Flask [106]	- Mainly for web apps - Workflows defined in Python
Data Factory [107]	- Commercial and quite expensive - Native compatible with othe Azure apps
Azkaban [108]	- A batch workflow job scheduler created at LinkedIn to run Hadoop jobs - e.g. like Oozie in many ways
Oozie [109]	- Hadoop world workflow automation (distributed) - In Cloudera, Hortonworks distros. Maby in AVA too? - Runs different types of actions: java, shell, spark, hadoop, etc. - Jobs uses xml to config the workflow graphs

References

1. SEMANTIC Proposal document, table 1.1.a – SEMANTIC research challenges, ESR assignment, 5GPPP performance KPIs, SEMANTIC KPIs and target values, page 7.
2. https://www.mycom-osi.com/hubfs/Ebook_Use%20Cases%20Q319/UC03%20-%20Simplifying%20RAN%20optimization%20for%20successful%20LTE%20deployments%20-%20Q319.pdf
3. https://www.subtonomy.com/
4. https://www.polystar.com/products/monitoring-osix/
5. https://www.splunk.com/en_us/platform.html
6. https://www.manageengine.com/network-monitoring/what-is-snmp.html
7. https://www.juniper.net/us/en/products-services/network-automation/paragon-active-
8. https://netbox.readthedocs.io/en/stable/
9. https://github.com/saltstack/salt
10. https://software.cisco.com/download/home/286314745/type/286283941/release/4.4.4
11. https://www.netscout.com/product/truecall#benefits
12. https://go.redhat.com/delivery-with-ansible-20181012?sc_cid=7013a000002DTD7AAO
13. https://www.teradata.com/Blogs/Streaming-Data-Into-Teradata-Vantage-Using-AWS-Glue-Streaming-ETL
14. https://kafka.apache.org/downloads
15. https://www.elastic.co/what-is/elasticsearch
16. https://aws.amazon.com/kinesis/
17. https://developers.google.com/analytics/devguides/platform
18. https://github.com/openshift/okd#getting-started
19. https://www.jaegertracing.io/docs/1.22/monitoring/
20. https://www.ibm.com/products/infosphere-datastage
21. https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-quick-create-portal
22. https://cloudify.co/
23. https://www.netscout.com/arbor-ddos
24. https://www.ibm.com/products/workload-automation#
25. https://www.logicmonitor.com/network-monitoring
26. https://github.com/aws-cloudformation
27. https://github.com/chaosblade-io/chaosblade
28. https://www.dynatrace.com/platform/
29. https://www.bluedanube.com/products/
30. https://www.exfo.com/en/products/service-assurance-platform/nova-fiber/
31. https://github.com/cisco/cisco-network-chef-cookbook?files=1
32. https://www.section.io/paas-hosting-providers-edge-solutions/
33. https://www.section.io/saas-edge-solutions/
34. https://github.com/hashicorp/terraform
35. https://aws.amazon.com/cloudwatch/
36. https://github.com/chaosblade-io/chaosblade
37. https://www.ibm.com/products/infosphere-datastage
38. https://chartmogul.com/product/analytics/
39. https://gec.io/industrial-solutions/industrial-edge-appliance/
40. https://docs.microsoft.com/en-us/azure/automation/automation-dsc-getting-started
41. https://istio.io/latest/docs/setup/getting-started/
42. https://github.com/istio/istio
43. https://github.com/aws-cloudformation
44. Ghasemi, Amir. "DATA-DRIVEN PREDICTION OF CELLULAR NETWORKS COVERAGE: AN INTERPRETABLE MACHINE-LEARNING MODEL." In 2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP), pp. 604-608. IEEE, 2018. DOI: 10.1109/GlobalSIP.2018.8646338

45. Schwind, Anika, et al. "Crowdsourced Network Measurements in Germany: Mobile Internet Experience from End User Perspective." Broadband Coverage in Germany; 14. ITG Symposium. VDE, 2020.
46. Ahmed, Jawwad, Amadeu do Nascimento Júnior, Caner Kilinc, Dan Pan, Jaume Rius i Riu, and Jorgen Gustafsson. "Using Blackbox ML Techniques to Diagnose QoE Problems for an IPTV Service." In NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium, pp. 1-8. IEEE, 2020. DOI: 10.1109/NOMS47738.2020.9110375
47. Centralized vs Decentralized Machine Learning Approaches for Proactive Service Assurance of a Commercial IPTV Service
48. Benzaid, Chafika, and Tarik Taleb. "AI-driven zero touch network and service management in 5G and beyond: Challenges and research directions." IEEE Network 34, no. 2 (2020): 186-194., DOI: 10.1109/MNET.001.1900252.
49. Khanafer, Rana M., Beatriz Solana, Jordi Triola, Raquel Barco, Lars Moltsen, Zwi Altman, and Pedro Lazaro. "Automated diagnosis for UMTS networks using Bayesian network approach." IEEE Transactions on vehicular technology 57, no. 4 (2008): 2451-2461, DOI: 10.1109/TVT.2007.912610.
50. Diaz-Aviles, E., Pinelli, F., Lynch, K., Nabi, Z., Gkoufas, Y., Bouillet, E., Calabrese, F., Coughlan, E., Holland, P. and Salzwedel, J., 2015, October. Towards real-time customer experience prediction for telecommunication operators. In 2015 IEEE International Conference on Big Data (Big Data) (pp. 1063-1072). IEEE. DOI: 10.1109/BigData.2015.7363860.
51. Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. The elements of statistical learning. Vol. 1, no. 10. New York: Springer series in statistics, 2001.
52. Campello, Ricardo JGB, Peer Kröger, Jörg Sander, and Arthur Zimek. "Density-based clustering." *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 10, no. 2 (2020): e1343. DOI/abs/10.1002/widm.30
53. Temprado, Y., Molinero, F. J., Garcia, C., & Gomez, J. (2008, December). Knowledge discovery from trouble ticketing reports in a large telecommunication company. In 2008 International Conference on Computational Intelligence for Modelling Control & Automation (pp. 37-42). IEEE. DOI: 10.1109/CIMCA.2008.116
54. Conference on Computational Intelligence for Modelling Control & Automation, pp. 37-42. IEEE, 2008.
55. Trouble Ticket Resolution. In The Thirty-First International Flairs Conference.
56. https://wiki.o-ran-sc.org/
57. https://www.nokia.com/networks/products/airframe-open-edge-server/
58. https://www.nokia.com/networks/solutions/airscale-cloud-ran/
59. https://www.ansible.com/
60. https://www.nokia.com/networks/solutions/cloudband/
61. https://www.nokia.com/networks/products/deepfield-cloud-intelligence/
62. https://github.com/nokia/helm-charts/tree/master/stable/logstash
63. https://docs.tenable.com/nessus/Content/CloudServices.htm
64. https://www.nokia.com/networks/solutions/netguard-xdr-security-operations
65. https://www.nokia.com/networks/solutions/openstack-private-cloud-for-large-enterprises/
66. https://www.zabbix.com/
67. https://www.nokia.com/networks/solutions/airframe-data-center-solution/
68. https://www.nokia.com/networks/solutions/netact/
69. https://www.nokia.com/networks/solutions/flowone/
70. https://www.nokia.com/networks/products/deepfield-defender/
71. https://www.nokia.com/networks/products/deepfield-operational-intelligence/
72. https://www.nokia.com/networks/solutions/netact/
73. https://www.nokia.com/networks/5g/ip/
74. https://powerbi.microsoft.com/en-us/
75. https://www.infovista.com/tems/pocket
76. https://vulcan.io/
77. https://aws.amazon.com/
78. https://azure.microsoft.com/en-us/

79. https://confluence.int.net.nokia.com/display/K2/K2+Machine+Learning+Home [internal]
80. https://nokia.sharepoint.com/sites/AVA/Support/Home.aspx [internal]
81. https://mlflow.org/
82. https://mlflow.org/
83. https://valohai.com/
84. https://hortonworks.com/
85. https://www.cloudera.com/
86. https://software.intel.com/en-us/openvino-toolkit
87. https://developer.nvidia.com/tensorrt
88. https://www.tensorflow.org/tfx
89. https://confluence.app.alcatel-lucent.com/display/plateng/CDLK+-+Analytics+Data+Lake [internal]
90. https://www.acumos.org/
91. https://scikit-learn.org/stable/
92. https://pytorch.org/
93. https://www.tensorflow.org/tfx
94. https://caffe.berkeleyvision.org/
95. http://deeplearning.net/software/theano/
96. https://spark.apache.org/mllib/
97. https://deeplearning4j.org/
98. https://github.com/spotify/luigi
99. https://airflow.apache.org/
100. https://argoproj.github.io/argo/
101. https://brigade.sh/
102. https://blog.fission.io/posts/kafka_sample/
103. https://www.spinnaker.io/concepts/pipelines/
104. https://www.kubeflow.org/
105. http://pydoit.org/
106. http://flask.pocoo.org/
107. https://azure.microsoft.com/en-us/services/data-factory/
108. https://azkaban.github.io/
109. http://oozie.apache.org/
110. https://caffe.berkeleyvision.org/
111. Y. Liu, J. Ding and X. Liu, "A Constrained Reinforcement Learning Based Approach for Network Slicing," 2020 IEEE 28th International Conference on Network Protocols (ICNP), 2020, pp. 1-6. DOI: 10.1109/ICNP49622.2020.9259378
112. H. Halabian, "Distributed Resource Allocation Optimization in 5G Virtualized Networks," in IEEE Journal on Selected Areas in Communications, vol. 37, no. 3, pp. 627-642, March 2019. DOI: 10.1109/JSAC.2019.2894305
113. Q. Liu, T. Han, N. Zhang and Y. Wang, "DeepSlicing: Deep Reinforcement Learning Assisted Resource Allocation for Network Slicing," GLOBECOM 2020 - 2020 IEEE Global Communications Conference, 2020, pp. 1-6. DOI: arXiv:2008.07614 [eess.SP]
114. H. H. Esmat and B. Lorenzo, "Deep Reinforcement Learning based Dynamic Edge/Fog Network Slicing," GLOBECOM 2020 - 2020 IEEE Global Communications Conference, 2020, pp. 1-6. DOI: 10.1109/GLOBECOM42002.2020.9322631
115. X. Cheng, Y. Wu, G. Min, A. Y. Zomaya and X. Fang, "Safeguard Network Slicing in 5G: A Learning Augmented Optimization Approach," in IEEE Journal on Selected Areas in Communications, vol. 38, no. 7, pp. 1600-1613, July 2020. DOI: 10.1109/JSAC.2020.2999696
116. Chaoyun Zhang, Marco Fiore, Cezary Ziemlicki, Paul Patras. Microscope: Mobile Service Traffic Decomposition for Network Slicing as a Service. The 26th ACM Annual International Conference on Mobile Computing and Networking (MobiCom ’20), Sep 2020, London, United Kingdom. DOI: 10.1145/3372224.3419195

117. D. Bega, M. Gramaglia, M. Fiore, A. Banchs and X. Costa-Perez, "DeepCog: Cognitive Network Management in Sliced 5G Networks with Deep Learning," IEEE INFOCOM 2019 - IEEE Conference on Computer Communications, 2019, pp. 280-288. DOI: 10.1109/INFOCOM.2019.8737488
118. Liu, Qiang and Tao Han. "DIRECT: Distributed Cross-Domain Resource Orchestration in Cellular Edge Computing." Proceedings of the Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing , July 2019, pp 181-190. DOI: 10.1145/3323679.3326516
119. Z. Yan, J. Ge, Y. Wu, L. Li and T. Li, "Automatic Virtual Network Embedding: A Deep Reinforcement Learning Approach With Graph Convolutional Networks," in IEEE Journal on Selected Areas in Communications, vol. 38, no. 6, pp. 1040-1057, June 2020. DOI: 10.1109/JSAC.2020.2986662