"From use cases to implementation-Machine Learning for Future Networks including 5G"

Vishnu Ram
ITU-T FG ML5G
Agenda

- Use cases
- Overall architecture framework
- Data handling framework
- Framework for evaluation of intelligence level
- Integration of ML marketplace
- ML Function orchestration
- Some Proof of concepts – student projects with FG.
- Future work: gaps and relations and liaisons
Use cases and requirements for ML in IMT-2020 and future networks

• More than 30 use cases submitted to the FG
• Requirements were analyzed for each, reviewed and compiled in ML5G-I-153.
• Requirements are classified as “critical”, “expected” and “added value”.
• Submitted to SG13
Use cases and requirements for ML in IMT-2020 and future networks

<table>
<thead>
<tr>
<th>Title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traffic Classification</td>
<td>collect a large amount of traffic data and learn the patterns of the collected data to build traffic classification models</td>
</tr>
<tr>
<td>Mobility Pattern Prediction</td>
<td>to collect position estimates for coarse grain, large scale and finer grain UE mobility.</td>
</tr>
<tr>
<td>Cognitive Heterogeneous Networks</td>
<td>ML in Cognitive Heterogeneous Networks allow allocation of resources from different communication networks access nodes</td>
</tr>
</tbody>
</table>
# Use cases and requirements for ML in IMT-2020 and future networks

<table>
<thead>
<tr>
<th>Title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML based Self Optimizing Networks (SON)</td>
<td>SON framework that integrates ML, provides a solution for the global network optimization problem, Supports multiple levels of the network and different technologies.</td>
</tr>
<tr>
<td>Network slicing</td>
<td>analyses the complex slicing requirements and the radio network conditions, and fulfills the requirements of each slice while achieving optimal radio resource utilization.</td>
</tr>
<tr>
<td>Edge intelligence</td>
<td>ML architecture with controllers implemented in the edge to collect the data generated by the network, run analytics and extract relevant metrics.</td>
</tr>
</tbody>
</table>

Also use cases for Quality of Experience (QoE) optimizations, Industry 4.0, network automation, Root cause analysis (RCA), network management, link adaptation, etc.
“Learnings” from the use cases

Declarative approach to designing and using ML in the network

- Define the ML use case separately from the underlying technology.
- Instantiation of the ML on the underlying technology can use technology specific interfaces (or extensions).

ORCH: Orchestrator
“Learnings” from the use cases

Evolve the ML “layer” independent of the underlying technology evolution.

- Map the technology-specific data model (DM’) to a generic (umbrella model, DM) used in the ML intent.
- Select and train models based on the generic data model.

Repo: Repository
DM: Data model
ORCH: Orchestrator

ITU-T ML pipeline
“Learnings” from the use cases

Evaluate and negotiate ML capabilities

- Multi-vendor, multi-level technologies will be used in future.
- ML use case spanning multiple levels will need compatible intelligence levels.

ITU-T ML pipeline
Architecture framework (ITU-T Y.3172)

• 20 architectural requirements

• Derived from use cases

• Classified into categories:
  • Enablers for correlation of data across levels and heterogeneous technologies
  • Enablers for deployment
  • Requirements related to interfaces between the architectural components
  • Requirements related to declarative specifications used for specifying the ML applications
  • Requirements related to the management of the architectural components

• Components of architecture framework for ML in IMT-2020 are laid out
• High level architecture is described
# Architecture framework (requirements)

<table>
<thead>
<tr>
<th>Requirement ID</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REQ-ML-SPEC-001</td>
<td>support a standard method to represent ML applications, which can be translated into ML functionalities in technology-specific underlay network functions.</td>
</tr>
<tr>
<td>REQ-ML-COR-003</td>
<td>The ML architecture is required to support distributed instantiation of machine learning functionalities in multiple levels.</td>
</tr>
<tr>
<td>REQ-ML-DEP-003</td>
<td>support flexible placement of ML functionalities (in coordination with the management and orchestration functions in the underlying network functions.</td>
</tr>
<tr>
<td>REQ-ML-DEP-004</td>
<td>support plugging in and out new data sources or configuration targets to a running ML environment.</td>
</tr>
</tbody>
</table>
Architecture framework (components)

MLFO: Machine learning function orchestrator

ITU-T ML pipeline
Architecture framework

- SRC: source
- C: Collector
- PP: Preprocessor
- M: Model
- P: policy
- D: Distributor
- NF: Network Function

ITU-T ML pipeline
Framework for data handling to enable ML in future networks

ML5G-I-148

Qi Sun (China Mobile)  
Yaxing Xu (China Mobile)  
Yuxuan Xie (China Mobile)  
Liya Yuan (ZTE)  
Sihai Zhang (USTC)  
Salih Ergut (Turkcell)
ML Use case: Cognitive, self-healing, heterogeneous networks (ML5G-I-153)

Requirements:
- Dynamic resource allocation in different parts of the network based on ML output.
- ML data collection and application of ML output may be done from various levels in the network.
- Flexible hosting of deep learning models

Data Collection:
- From plug-and-play configuration of newly deployed NFs
- ML data collection and ML data output should support different wireless technologies of different generation.

- ML data output may be fully automated or semi-automated.
Challenges for data handling framework

• Various components in the network produce data with differing characteristics.
• Flexibility in network configurations, dynamically evolving sources of data and applicable network configuration parameter and policy.

• 90 requirements for data handling captured and analysed.
• Classified into
  - ML data collection
  - ML processing
  - ML data output

• Framework for data handling is defined based on these requirements

• Sequence diagrams for each scenario is explained
  • Figure 8.2 Deployment of data handling framework
  • Figure 8.3 Addition of new SRC in data handling framework
  • Figure 8.4 Data model does not exist in the ML metadata store in the data handling framework
Data handling framework

ITU-T ML pipeline
Acronyms

AF-s: Application Function (simulated)
API-g: API generic
API-s: API specific
DBr: data broker
DBrCP: data broker control plane
DBrUP: data broker user plane
DM: data model
IETF: Internet Engineering Task Force
NF: Network Function
NF-s: Network Function (simulated)
SDO: standard development forum
TMF: TM Forum
Data handling framework

1. ML Intent
   2. DM Lookup req
   3. DM Lookup resp
   4. Create DBr session

Sub procedure: Create ML pipeline

5. Provision DM
   6. Create DBr-UP instance
   7. Provision DM resp
   8. Provision DM
   9. Create DBr-UP instance
10. Provision DM resp
11. Provision DM
12. Provision DM resp
13. Create DBr session response
   14. Data (via API-s)
   15. Data (via API-g)
   16. Data storage
   17. ML Output (via API-g)
   18. ML output (via API-s)

ITU-T ML pipeline
Method for evaluating mobile network intelligence level

ML5G-I-151

Cao Xi, China Mobile
Wang Liang, ZTE
Ni Hua, ZTE
Hu Bo, Beijing University of Posts &Telecom
Saliha Sezgin Alp, Turkcell
Salih Ergüt, Turkcell
Tarık Kranda, P.I. Works
Dharmendra Misra, Consultant, IBM
Yameng Li, China Unicom
Xiaoqing Xu, China Telecom
Significance of evaluating intelligence

• Provide evaluation basis for measuring the intelligent capability level of a mobile network and its components.

• Common understanding with equipment vendors.

• Help the industry to reach a consensus of the concepts of intelligent mobile network.

• Provide reference for to formulate relevant strategies and development planning of future mobile networks.

• Provide decision support for planning for products roadmaps.
Step-1: Dimensions for Evaluating the Intelligence Level of Mobile Networks

**Workflow Intelligence**
- Planning and design
- Deployment
- O&M and optimization
- ...
- Service providing

**Dimensions**
- Demand mapping
- Data collection
- Analysis
- Decision
- Action implementation

**Subsystem Intelligence**
- Network elements
- Management system
- Network platform
Step-2: Basic method for evaluating mobile network intelligence level

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Human</th>
<th>Human and System</th>
<th>System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Action implementation</td>
<td>completed fully by human.</td>
<td>completed automatically by the system in at least one scenario</td>
<td>completed automatically by the system in <strong>all</strong> the scenarios.</td>
</tr>
<tr>
<td>Data collection</td>
<td>completed fully by human</td>
<td><strong>By system</strong> according to <strong>Human</strong> defined rules</td>
<td>by the system according to <strong>system</strong> defined data collecting rules.</td>
</tr>
<tr>
<td>Analysis</td>
<td>completed fully by human</td>
<td>..</td>
<td>by the system according to <strong>system</strong> defined/learned rules</td>
</tr>
<tr>
<td>Decision</td>
<td>completed fully by human</td>
<td>..</td>
<td>by the system according to <strong>system</strong> defined/learned rules</td>
</tr>
<tr>
<td>Demand mapping</td>
<td>completed fully by human</td>
<td><strong>By system</strong> according to <strong>human</strong> defined template</td>
<td>by the system according to <strong>intention</strong>.</td>
</tr>
</tbody>
</table>
Step-3: Method for Evaluating Mobile Network Intelligence Level

<table>
<thead>
<tr>
<th>Level/Name</th>
<th>Action implementation</th>
<th>Data Collection</th>
<th>Analysis</th>
<th>Decision</th>
<th>Demand Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>L0 Manual operation network</td>
<td>Human</td>
<td>Human</td>
<td>Human</td>
<td>Human</td>
<td>Human</td>
</tr>
<tr>
<td>L1 Assisted operation network</td>
<td>Human &amp; System</td>
<td>Human &amp; System</td>
<td>Human</td>
<td>Human</td>
<td>Human</td>
</tr>
<tr>
<td>L2 Primary intelligence</td>
<td>System</td>
<td>Human &amp; System</td>
<td>Human &amp; System</td>
<td>Human</td>
<td>Human</td>
</tr>
<tr>
<td>L3 Intermediate intelligence</td>
<td>System</td>
<td>System</td>
<td>Human &amp; System</td>
<td>Human &amp; System</td>
<td>Human</td>
</tr>
<tr>
<td>L4 Advanced intelligence</td>
<td>System</td>
<td>System</td>
<td>System</td>
<td>System</td>
<td>Human &amp; System</td>
</tr>
<tr>
<td>L5 Full intelligence</td>
<td>System</td>
<td>System</td>
<td>System</td>
<td>System</td>
<td>System</td>
</tr>
</tbody>
</table>
Evaluation: Use cases and examples

- Requirements are listed based on the scheme presented

- Use cases:
  - Network coverage optimization
  - Network cutover
  - RCA
  - Resource maintenance and management
  - E2E IoT service for utilities

- Architecture components and interfaces are listed.
ML Marketplace requirements and architecture

ML5G-I-146

Tengfei Liu, China Unicom
Yongsheng Liu, China Unicom
ML marketplace is hot!!

AI GLOBAL

Synapse AI
Decentralized Data and AI Marketplace

Easy-to-use APIs

ITU-T ML pipeline

LF Deep Learning Delivers First Acumos AI Release Making it Easier to Deploy and Share Artificial Intelligence Models
ML Use case: Root Cause Analysis, SON, network optimization, ... (ML5G-I-153)

Requirements:
- Using ML to identify the root cause of problems in the network.
- Prediction of high load scenarios based on usage
- Recommendations on optimizations in the network (e.g. scale down)

Data Collection:
- From Access Network, Core Network, Management Plane, ..
- Logs, events, alarms, configurations, state data, mobility patterns.

Analysis:
- Deep learning mechanisms, trained in the sandbox.

ML Output:
- Prediction on maintenance, decisions on resource scaling, isolation of errors.
Amounts of device alarms of IPRAN: IPRAN is the local integrated carrier transmission network of China Unicom. IPRAN is mainly used for 4G mobile service, and VIP customer service, and IPRAN uses IP/MPLS dynamic protocol. Compared with the traditional network, the protocol used by IPRAN is relatively complex, and the logical connection of the network is complicated. Compared with the traditional network management system, the IPRAN network system receives a large number of device alarms, many of which are caused by the root alarm.

The current processing method: For the alarms, the current method is to solve the alarms depending on the expert experience, which means transforming the expert experiences into rules, and filter out the non-critical alarm through the rules. The downside of this approach is that in order to avoid filtering out the important alarms, the filter rules are relatively relaxed, which means the rules have limited filtering ability.

It is hoped that applying AI to trace the root alarm can form a more efficient solution.
ML Use case: Technical Scheme and Process

Step One: Data preparation
- Identifying frequent alarms

Step Two: Correlation rules mining
Improving PrefixSpan algorithm

Step Three: Alarm correlation rules confirmation and storing
- Correlation rules
- Blacklist

Step Four: Root alarm identification
- Labelling alarms

Fig.1 Framework of technology scheme
ML Use case: Root Cause Analysis, SON, network optimization, ... (ML5G-I-153)
Acronyms:
AF: Application Function
ASF: Authentication Server Function
CEF: Capability Exposure Function
IPRAN: IP Ran access networks
MPLS: Multiprotocol label switching
NACF: Network Access Control Function
NFR: Network Function Repository
NSSF: Network Slice Selection Function
PCF: Policy Control Function
UPF: User Plane Function
SMF: Session Management Function
USM: Unified Subscription Management
Motivations for ML marketplace integration

• Interoperability between the mechanisms for ML marketplace is a challenge for network operators.

• Network operators should be able to follow the innovation curve in the ML marketplace.

• Standard mechanisms for efficient transfer and handling of ML models.

• Efficient use of testing methods requires sharing of information between ML marketplaces and network operators.

• ML models hosted in ML marketplace need to be amenable to the deployment methods used by operators.
Requirements for ML marketplace integration

REQ-ML-MKT-001: ML marketplace is required to support and host untrained models which can be trained offline, from stored data.

REQ-ML-MKT-002: ML marketplace is required to support indication of the maturity of testing status of the ML models.

REQ-ML-MKT-003: ML marketplace is required to support rich metadata about the ML models.

REQ-ML-MKT-007: ML marketplace is required to indicate the type of functions which may be applied on the ML model.

REQ-ML-MKT-008: ML marketplace is required to allow search and query of ML models based on specific characteristics.

REQ-ML-MKT-010: ML marketplace is recommended to allow import and export of ML models.
Architectures for ML marketplace integration

- External ML Marketplace
- Internal ML Marketplace
- ML pipeline
- ML underlay networks

Management subsystem:
- Other management and orchestration functions
- MLFO
- ML intent

ITU-T ML pipeline
ML Market place: training flow

ML intent → MLFO

Select Model, ML pipeline nodes

Provision data pipeline

Select training data

Provision training pipeline

Provision SRC, SINK

ML pipeline in ML sandbox

Simulated ML underlay networks

SRC is configured to generate data

SINK is configured to use ML output

ITU-T ML pipeline
ML Market place: monitor flow

- **ML intent**
- **MLFO**
- **Parse and decide the monitoring requirement**
- **Provision monitoring event**
- **Simulated ML underlay networks**
- **Generate periodic monitoring report**
- **Monitoring Report**
- **Handle monitoring report and provide output report**

**Flow Diagram**

- ML pipeline in ML sandbox
- Generate periodic monitoring report
- Monitoring Report
- Handle monitoring report and provide output report
ML Market place: serving flow

- MLFO
- ML pipeline in ML sandbox
- ML underlay networks

- Monitor flow
- Provision ML pipeline
- Transfer learning request
- Transfer learning

Evaluate the maturity of ML pipeline

ITU-T ML pipeline
ML Market place: retraining flow

- MLFO
- ML pipeline in ML sandbox
- Simulated ML underlay networks

Monitor flow

Plugin: Optimization Algo

Modify data plugin

Modify training plugin

Modify SRC and SINK

data

config
MLFO requirements and architecture

ML5G-I-150
Shagufta Henna,
Telecommunication Software & Systems Group, Waterford Institute of Technology
MLFO requirements and architecture

• MLFO is a logical orchestrator that can monitor and manage the nodes in a machine learning pipeline.

• Optimal placement of ML pipeline nodes is achieved in the network with the help of MLFO, coordinated with management plane of the ML underlay networks.

• MLFO lets the network operator specify the ML application using a declarative specification, abstracted from the ML underlay networks.

• Chaining and split of ML pipeline nodes, selection of ML models, monitoring their performance, reselection and update, if needed, are achieved using MLFO.
Relations and future work

Fig 2: Ecosystem

ITU-T ML pipeline
Liaisons

point 1: **Richer metadata** + standard machine learning model representation mechanisms.

point 2: **Chaining**+ with respect to the unique requirements of the edge.

point 3: **distributed training**: what are

point 4: **optimizations at run time**: to provide feedback to operator.
Student projects: PoCs

FG is offering guidance to uni students for doing relevant projects.

List of projects is described in ML5G-I-144

**How to join:**

fgml5g-students@lists.itu.int
Vishnu.n@ieee.org

• 15-20 students actively contributing at any point of time

• Across 4-5 countries
Student projects: PoCs

fgml5g-students@lists.itu.int
Thank you!
(vishnu.n@ieee.org)