The explorations and challenges for AI based Fault Prediction and Prevention on ICT system

Pei Ke
Corporate Reliability Dept of Huawei 2012 Lab
peiKe@huawei.com
2018.10
Huawei Reliability Department Introduction

- Dedicated **team**, Advanced **technologies and solutions POC, standard, E2E process,**
- To **break the silos** between products, between reliability and product R&D engineers.
Scope of Huawei Reliability

Service-level reliability
• QoS/QoE/Performance
• SLA

Network-level reliability
• Network design/Geo-Redundancy
• Traffic control/ load balance
• Survivability /Disaster Recovery

Equipment-level reliability
• Software reliability
• Hardware Reliability
• Mechanical Reliability

Technology, Methodology, Architecture, Design, Tools,......
Agenda

- Background
- Methodology of intelligent fault management
- The challenges and exploration
3 Levels of “All Cloud” Evolution for Telecom Industry

Virtualization

- Hardware and software decoupled

Cloud Native

- 5G oriented function decomposition
- Programmable
- DevOps & JAD: service level SW publish, A/B Test, chaos monkey
- Open platform for developers

Cloudification

- Service Governance framework
- Stateless Design
- N-way & Cross-DC Geo Redundancy
- Graceful Scalability
- Openness with 3rd party integration

Orchestration

- Focus on SW/HW decouple

Focus on Elasticity & Resilience

- Static Slicing: MVNO, IOT, ESN...
- CloudB2B: Cloud CDN, Cloud UC, CloudVPN

Focus on Business Agility

- Dynamic Slicing: AR/VR, V2X, Industry 4.0
- Market place for millions of apps
The Reality of Public Cloud Reliability

- Google Compute Engine, 2017-01-30, 4hrs
- AWS's S3 outage, 2017-02-28, 4hrs
- Facebook, 2017-02-24, 3hrs
- Microsoft Azure Storage loses power for eight hours due to "software error", 9hrs, 2017-03-16
- Microsoft Office 365, 17hrs, 2017-03-21,
- Apple's iCloud backup outage, 2 days, 2017-06-28

Actually there are much more outages happened, far more the ones list here.

Source: CloudHarmony

Apple to Apple? The Myth of 9s —— E2E? Real time service? Partial Outages?......
How to improve it?

• MTBF (Mean Time Between Failures): \( MTBF = \frac{\sum[(\text{downtime})_n - (\text{uptime})_{n-1}]}{\text{number of failures}} \)

• MTTR (Mean Time To Repair): \( MTTR = \frac{\sum[(\text{uptime})_n - (\text{downtime})_n]}{\text{number of failures}} \)

• A (Availability) = \( \frac{MTBF}{MTBF + MTTR} = \frac{\text{total time} - \sum(\text{time to repair}) + \text{number of failures}}{\text{total time}} \)

The smaller, the better

Reduce failover time with quick diagnostic recovery

Reduce the probability of failure by early detection and prevention of failures

Time Between Failures = { down time - up time}
Idea: From Fire-Extinguishing to Fire-Prevention

Health status and maintenance flow

- Normal state
- Subhealthy
- Faulty state

Past

- Services are affected.
- User complaint

Future

- Predict and prevent faults, identify potential risks, and eliminate potential risks.

Reactive approaches

- No prevention cost
- but prolonged service downtime

Proactive approaches

- Provide better system reliability
- but incur large overhead

Move the maintenance point forward.
To detect sub-health status and take right action to avoid customer impact.
Our Vision

Sub health: equipment or network is in sub-health state, but business is in normal range.

Problem acceptance analysis

Abnormal perception

Failure

Normal status

Level 1: Data pipeline is ready or not
Level 2: AI knows abnormal or not?
Level 3: AI knows “what happened?”
Level 4: AI can knows ”what will happen?”
Level 5: AI can suggest”what action need to be taken?”, which are carried out manually.
Level 6: full automation enables self-healing.
FPP Evolution path
FPP Evolution path
FPP Evolution path
FPP Evolution path
Agenda

- Background
- Methodology of intelligent fault management
- The challenges and exploration
Three elements of AI project success

**Data:**
Algorithms without data are useless. Data is the core of the algorithm, so getting a lot of data will become the top priority.

**Algorithm:**
Google acquired DeepMind to gain competitive advantage. FB has acquired Wit.ai to enhance speech recognition and voice interface services.

**Computing power:**
- Google TPU: 2016.5
- NVIDIA Tesla P100 GPU: 2016.4
- Microsoft FPGA: 2016.9
- HW Ascent910/310: 2018.10
Gartner Hype Cycle state

Gartner think it’s in “Innovation trigger” stage from 2017 to now, no change, it needs five to ten years to mature. So pessimistic?

Source: Gartner Hype Cycle for artificial intelligence 2017

Source: Gartner, Hype Cycle for ICT in India, 2018
Industry players

Facebook's auto-remediation FBAR and Winston system by Netflix ⇒ no technical details available

- Near zero learning time
- Real-time analysis
- Handling unknown anomalies
- Streaming analysis
Methodology -- An Overview of FPP data pipe

Data suitability assessment & enhancement iteration loop

 rawDataPool

Data Acquisition

Feature Extraction

Detectability Assessment

Yes

Anomaly Detection

Yes

Fault Diagnosis

Yes

Degradation Assessment & RUL Prediction

Data Suitability Assessment

Diagnosability Assessment

Yes

Prognosability Assessment

No

Feature Selection

FeaSet for Fault Diagnosis

FeaSet for Fault Detection

FeaSet for Prognosis

Data suitability enhancement

Yes

Prognosability Assessment

No

Feature Extraction

Feature Selection

Detectability Assessment

Yes

Anomaly Detection

Yes

Fault Diagnosis

Yes

Degradation Assessment & RUL Prediction

No
Methodology – Data Exploration of FPP data pipe

Data from ICT system

<table>
<thead>
<tr>
<th>Health Level</th>
<th>100%</th>
<th>80%</th>
<th>60%</th>
<th>40%</th>
<th>20%</th>
<th>0%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Life Cycle #1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Life Cycle #2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Life Cycle #3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Data Files

Detectability

- **Step 1:**
  - **Healthy**
  - **Faulty**

- **Step 2:**
  - Feature Extractor
  - Sample-VS-Distribution (SvD)

- **Step 3:**
  - Healthy
  - Faulty
  - Distribution- VS-Distribution (DvD)

Diagnosability

- **Healthy**
- **Failure 1**
- **Failure 2**

Prognosability

- **Baseline Distribution**
- **Life Cycle Data**
- Health Value vs Time
  - Normal
  - In-between stage
  - Faulty
Algorithm Selection: The algorithm is widely used

Scenarios
- Health Prognosis
- Anomaly Detection
- Fault Diagnosis

Data pre-process
- Model Training
- Prediction
- Take Action

Mathmatic
- Clustering
- Regression
- Clarification
- TS analysis
- Correlation

Base
- SVM, ANN, HMM, IForest, RF, Boosting; K-Means, KNN, DBSCAN, SOM
- AR, MA, ARIMA, FFT, Wavelet
- Apriori, PCA, SVD, FP-Growth, Granger, Graph
- Decision Tree, RNN/LSTM, RL, PID

Platform
- Beam
- Spark
- TensorFlow
- Huawei Info Insight
- Huawei Mind
- ...
Case Study. NFV Cross Layer fault localization

Cross layer fault localization is a big challenge

- Prediction
- Detection
- Localization
- Recovery

Set of uncorrelated Fault events reported by distributed agents over predetermined time period $\Delta T$

Fault Pool

Dependency Model

Rule engine that determines the root causes from a chain of correlated faults for fault localization

Using dependency model to correlate fault events from related components

RCA
Case Study. NFV Cross Layer fault localization

Learn normal system state and identify deviations as anomalies

Root cause of the fault is judged according to the deviation degree of each operation environment factor

Accuracy: 0.96
Agenda

- Background
- Methodology of intelligent fault management
- The challenges and exploration
The real challenge is data

Garbage in, Garbage out: The quality and integrity of data is crucial for building an efficient model of AI

**Hard to acquisition**
- fewer commercial sites
- Data is **sensitive** and hard to access to analysis
- A small amount of fault data **can not be deposited for long time**

**Small data samples, unlabeled and unbalanced.**
- **labeled fault** data is difficult
- There are many problems type, but **few sample data** for each.
- The fault problem of existing network is complex, which requires a lot of resources to locate.

**Data format is not uniform**
- Data format is not uniform, each product data format up to 200+
- Data loss
- **Lack of standardization**
- Instrumentation incomplete
Less sample data - both simulated data and real data

- Simulated data combined with real network verification
  - Simulation of real network traffic tool
  - Fault Injection tool
  - NFV testbed and mirror env
- Successful case:
  - Slow disk detection
  - Wireless CPRI fault generation
  - Optical module failure
  - Memory leak failure
- The simulation of complex scenes remains to be improved

Diagram:
- Achieve user behavior diversification simulation
- Fault data simulation
- Labeled simulated fault data
- High quality data set
- Solution define
- Real Data from sites

Process:
- Fault data collection
- Fault injection
- Product testbed
- Traffic
- Fault data management
- Data Pool
- Data Factory
- Index platform
- Model lib
- Training
- DEV
- Validation
Few labeled data—semi supervised learning

- Idea: When labels are difficult to automatically acquire, it is usually labeled manually by a human oracle. Intuitively, randomly selecting instances to label experts is not the best strategy. Active Learning means asking Experts to label the selected “best value” sample.

In our experiments, we learned to automatically pick out the “most valuable” fault fingerprints and label them with domain experts. To achieve the same classifier accuracy, the number of samples required for active learning only needs one fifth or less of the traditional supervised learning.

Result in our case:
1. The more labeled samples, the lower the error rate.
2. No matter how many labels are selected, active learning is always better than traditional supervised learning.
3. The more samples are selected, the lower the cost performance.
4. The initial marker sample is 10, and the sample size of active learning is about 20.
Algorithm: AI+ expert experience are equally important

- Feature extraction is an important part of traditional ML and also DL, and algorithm performance depends on human experience.
  - There is a saying in the industry of ML: if the feature is not done well, tuning the parameter will never stop.

- Fault data labeling needs expert feedback confirmation
  - Labeling heavily depends expert knowledge
  - EAI: It can significantly reduce the risk of "intelligent mis-operation" in complex tasks.

- Expert knowledge is a very important asset in the weak AI stage, which needs better management and solidification.
  - Rule based case can cover 80%, only 20% for AI based
  - Case intelligent search and knowledge map
Relationship between model complexity and data

- Complex models are used to solve complex problems, and simple models solve simple problems.
- For limited data, simple models may be better than complex models for complex problems.
- Once data is enough, complex models can generate accurate results.
Hierarchical Intelligence /distributed ML:

- Intelligent Agent.
- Network Management Level Intelligence
- Cloud Intelligence

From data collection to abnormal perception, real-time local analysis of closed loop, reduce data transmission.
Challenge 1- algorithm model

- Challenges of updating system state change models: for example, upgrades, operational promotions, resulting in changes in KPI sample distribution, and increased failure types (model evolvable)

- Requiring model reuse in a similar environment where sufficient data is difficult to obtain: model reuse, transfer learning needed

Google infrastructure upgrades will evolve into continuous upgrades of the network, incremental upgrades. New features and configurations are pushed into the product every week, so upgrades will be made every day, even multiple times a day.

Note: 2016 Sigcom articles and 2017 blog from Google
Inter and Intra system Prediction Models transfer challenges

Inter DC model transfer

Intra DC model transfer
Challenge 2- development mode need be changed

Challenges to the existing development process: online data closed-loop algorithm model tuning, update. (flexible) - to collect data from the existing network to form iterative feedback.

Requirements for development environment: we need to combine the existing network data to optimize the model.
Summary

- AI can assist to improve system availability in some cases but not all.
- AI based fault predict and prevention presents many challenges, we long way to go.
- Domain expertise is very important in define the solution
- Data suitability analysis before model development is a must
- Define models should based on what data you have. Simple models may have good results.
Thank you

www.huawei.com