ITU AI/ML 5G Challenge

Theme 1 from KDDI

Analysis on route information failure in IP core networks by NFV-based test environment

UT-NakaoLab-AI Team

2020-10-15

Team Members: Fei Xia (M1), Aerman Tuerxun (M1), Jiaxing Lu (D1), Ping Du

Background

- Network failure and anomaly detection is desired to be automatically performed by AI/ML for operating 5G mobile networks.
- Challenges in AI/ML
  - How to extract features from the huge amount of unstructured log files collected from network routers and devices.
  - How to identify important features to reduce the computation.
  - How to detect failures with a small amount of data in real-time.

Summary

- First, we extract features from huge amount of log files into 997 features.
- Then, we refine features to with differential data to highlight the difference between normal and abnormal data.
- Third, we identify the most important 30 features so that we can reduce computation without degrading the performance.
- Moreover, our work extended KDDI's NOMS2020 paper as follows:
  - Our model can target 7 failure cases while the NOMS2020 paper address only 3 failure cases.
  - We can use use one unified model to predict both device and interface errors, while NOMS2020 paper use different models to predict network and network failures separately.
  - Besides the Multiple-layer perceptron (MLP), Random Forest (RF), and Support Vector Machine (SVM) machine learning algorithms, we also add Decision Tree (DT) and XGBoost (XGB) to the comparative analysis.

Agenda

01 Feature Extraction

02 Feature Refinement

03 Feature Reduction

04 Model Training

05 Model Evaluation

01

Feature Extraction

PART ONE

Feature Extraction

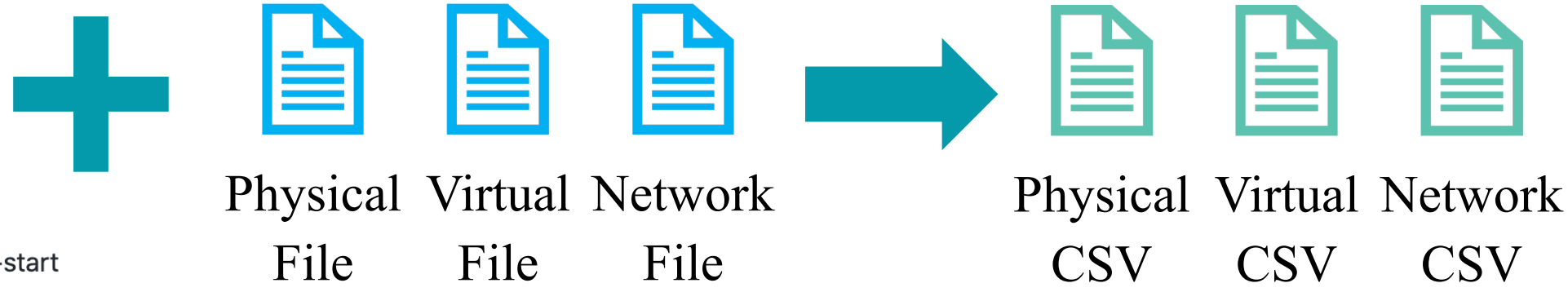
Extract features from Log files (JSON format) and merged features with label based on time it into CSV files

Log files

(14 Types of Events)

- 0: ixnetwork-traffic-start
- 1: node-down
- 2: node-up
- 3: interface-down
- 4: interface-up
- 5: tap-loss-start
- 6: tap-loss-stop
- 7: tap-delay-start
- 8: tap-delay-stop
- 9: ixnetwork-bgp-injection-start
- 10: ixnetwork-bgp-injection-stop
- 11: ixnetwork-bgp-hijacking-start
- 12: ixnetwork-bgp-hijacking-stop
- 13: ixnetwork-traffic-stop

Merge based on time



Key Points in Feature Extraction

- For all log files, we utilize path like "*key1/key1-1/key1-1-1…*" as keys to extract features from physical-infrastructure, virtual-infrastructure, and network-device JSON log files.

- For BGP related entries, we use the number of next-hops in each array and their prefixes as features.

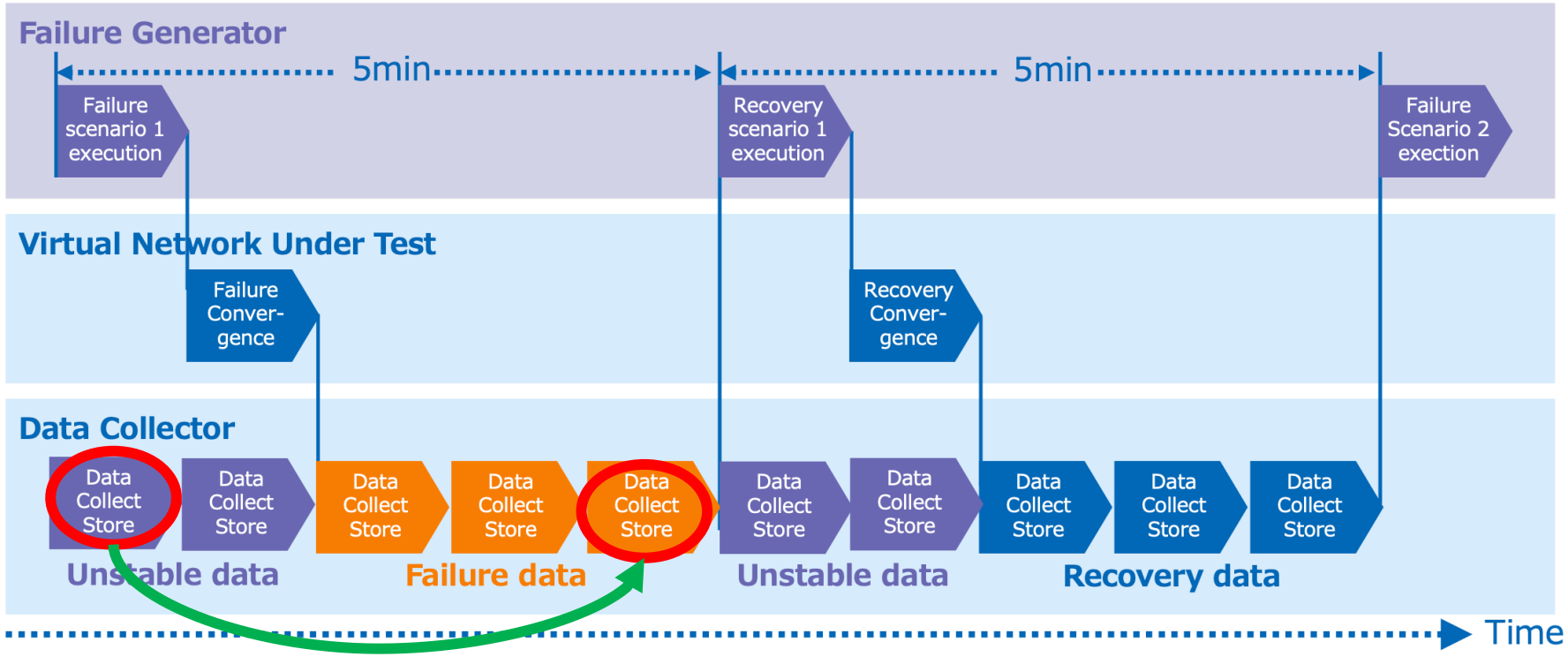
02

Feature Refinement

PART TWO

Differentiate Data as Input

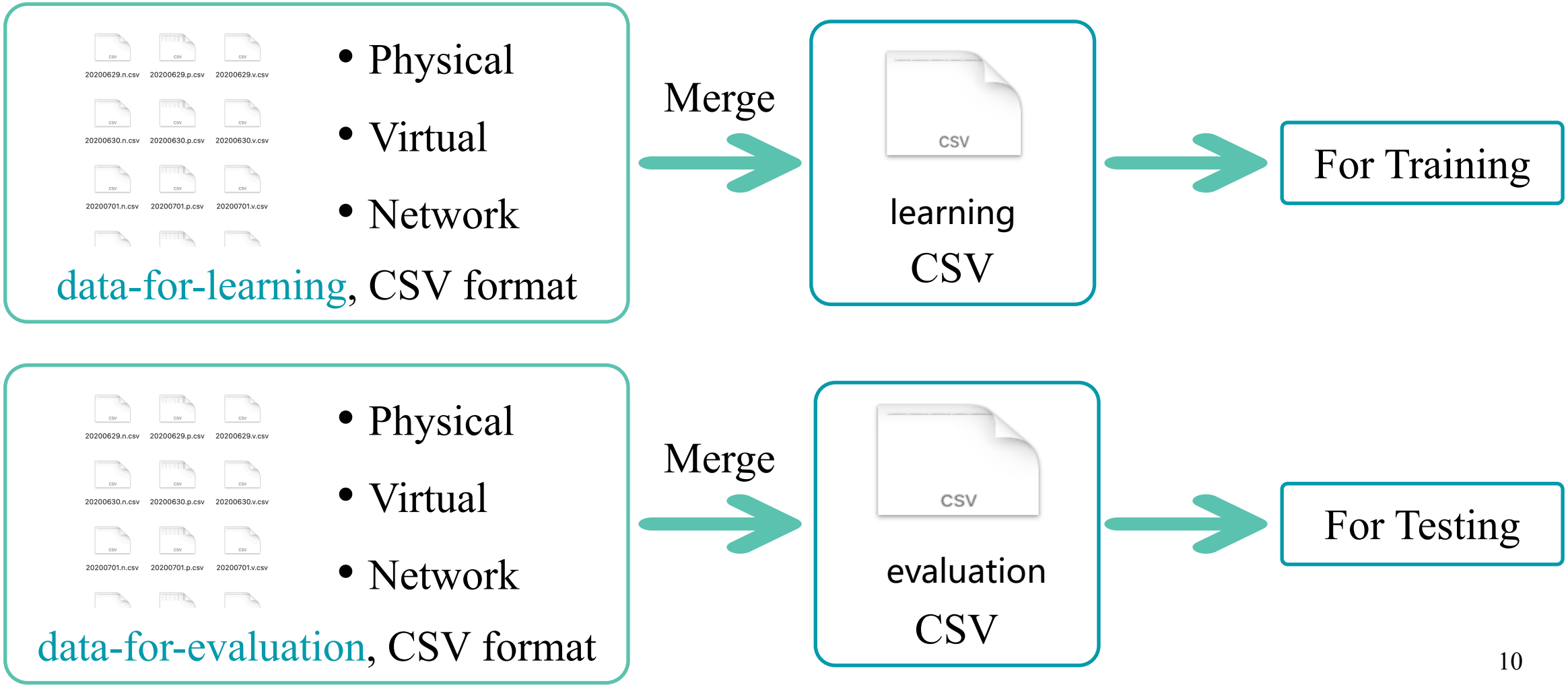
To highlight the difference between normal and abnormal data sets to derive metrics which have changed since the occurrence of a failure, we use differential data between the 5th minute data and the 1st minute data as input.



$$Diff\ Data = 5th\ Minute\ Data - 1st\ Minute\ Data$$

Merge diverse datasets

To train a unified model for diverse network events, we merge all datasets into one csv file for training and testing.



03

Feature Reduction

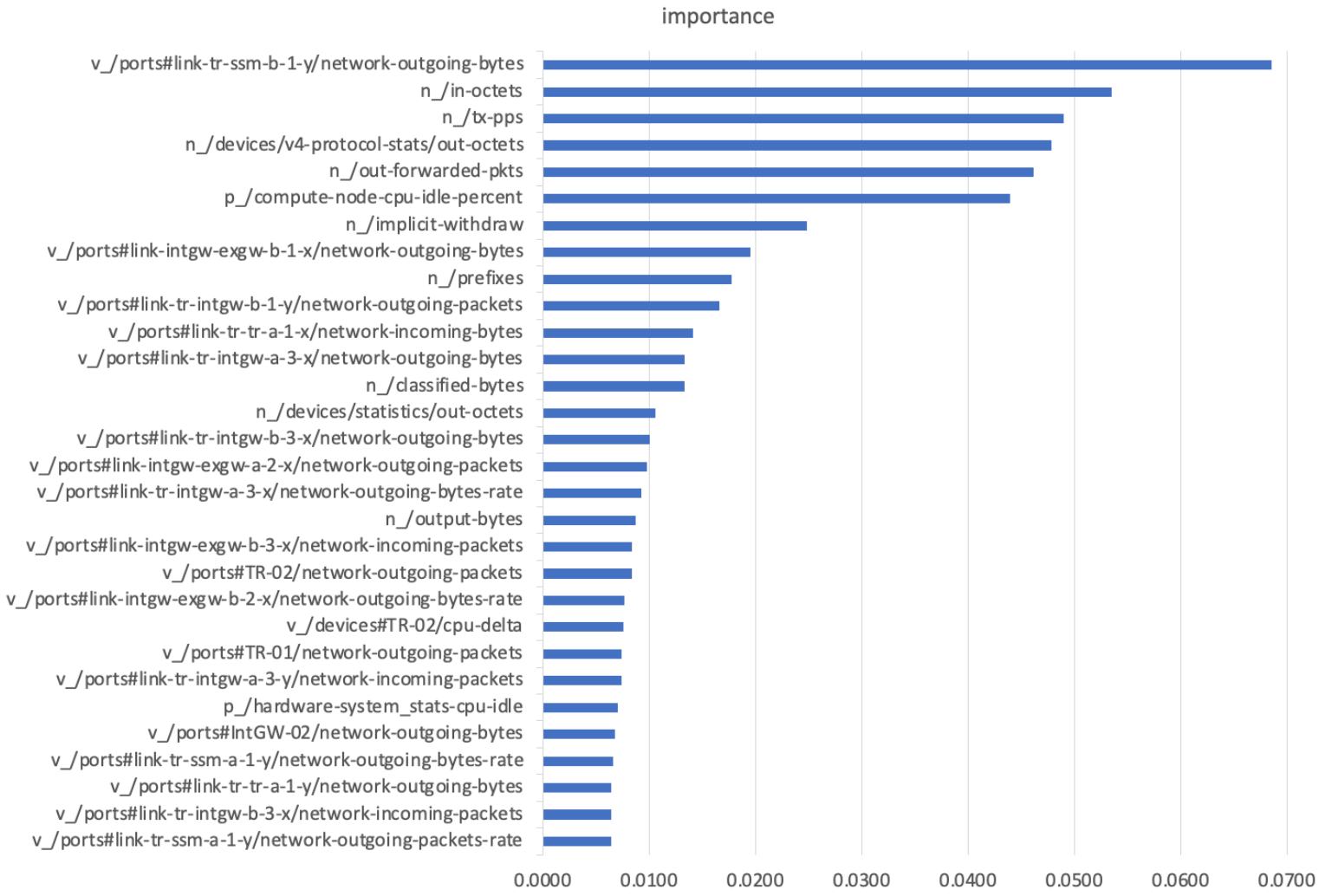
PART THREE

Feature Importance Analysis

Importance is calculated for a single decision tree by the amount that each attribute split point improves the performance measure, weighted by the number of observations the node is responsible for.

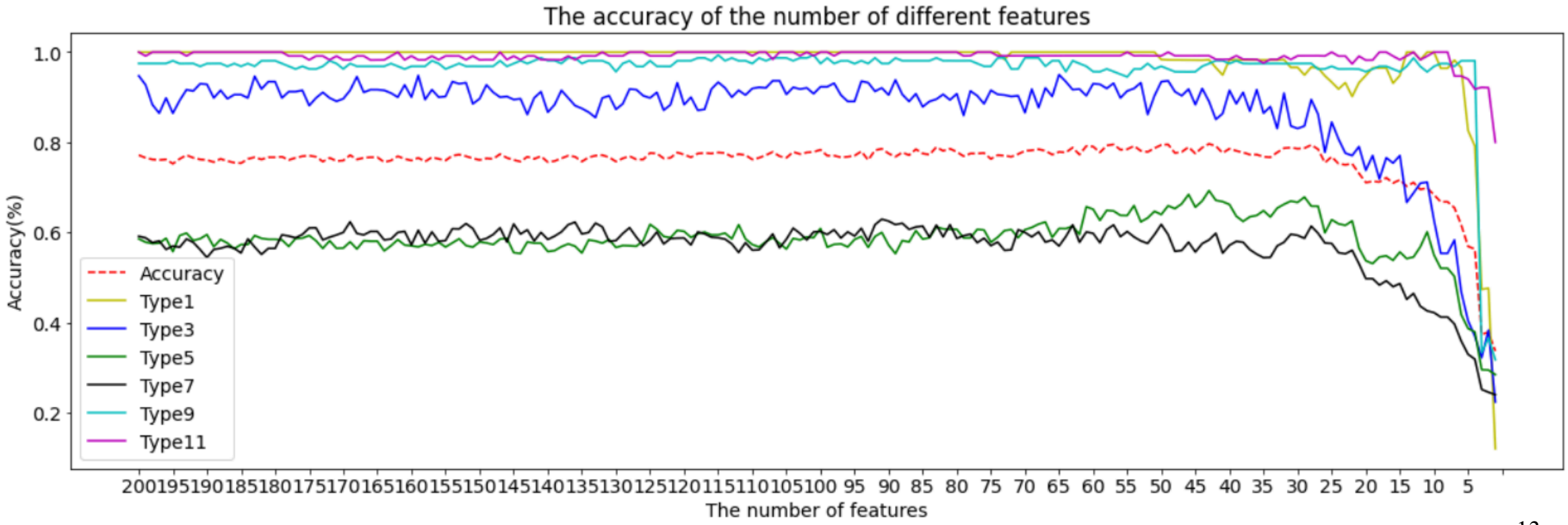
The performance measure may be the purity used to select the split points or another more specific error function.

The feature importance is then averaged across all of the the decision trees within the model.



Different Features Accuracy

- Use different numbers of features to train the data and observe the changes in accuracy.
  - Through importance analysis of features, we can only use top 30 important features to achieve the same high performance.

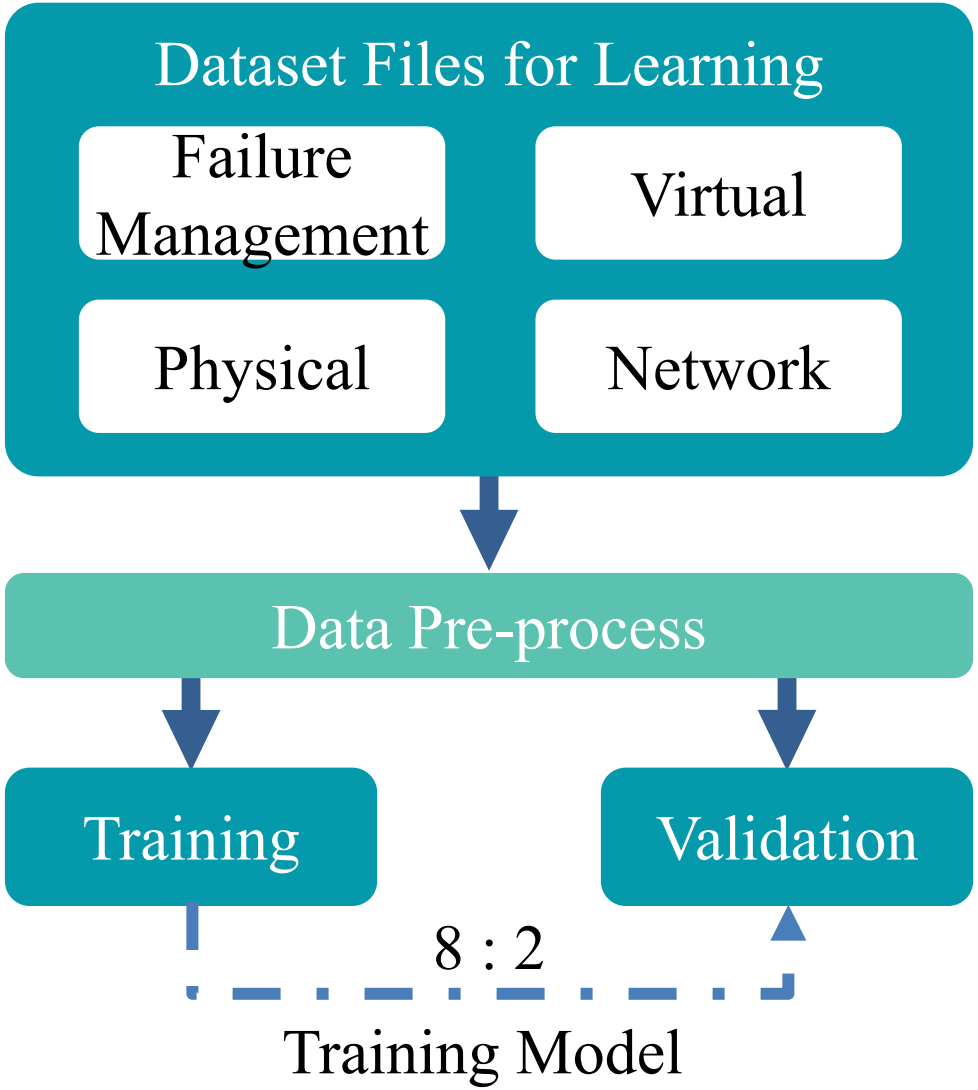


04

Model Training

PART FOUR

Training & Validation with Learning Data



No.	Method	Accuracy
1	XGBoost	0.77
2	Radom Forest	0.76
3	Decision Tree	0.73
4	SVM	0.58
5	MLP	0.54

Validation accuracy during training

05

Model Evaluation

PART FIVE

Evaluate By Precision

Type 1, 3, 9 and 11 are identified very well using most of the models while the accuracies of type 5 (Packet Loss) and 7 (Packet Delay) are very low

$$Precision = \frac{TP}{TP + FP}$$ (True Positive (TP), False Positive (FP))

	All Data		
Label Type	**dt**	**rf**	**xgb**
1: node-down	0.97	1.00	1.00
3: interface-down	0.41	0.97	0.85
5: tap-loss	0.45	0.56	0.54
7: tap-delay	0.40	0.50	0.55
9: ixnetwork-bgp-injection	0.97	0.99	0.97
11: ixnetwork-bgp-hijacking	0.84	1.00	1.00

Comparative Analysis

- Our comparative analysis confirm the KDDI's finding that Random Forest (RF) outperforms SVN and MLP algorithms.
- We also find that the XGBoost algorithm has a comparable performance as that of RF.
- We also found there is still room for improvement for the the failure cases: tap-loss and tap-delay, which are not discussed the NOMS2020 paper. These are left for our future work

Contributions

- First, we introduce our feature extraction mechanism especially for network BGP data
- Then, we use differential data as input to highlight features of abnormal data.
- Third, we analyze and identify the most important 30 features to reduce computation without degrading the performance.
- Comparing to the work in NOMS2020 paper:
  - Our model can target all list 7 failure cases while the NOMS2020 paper address only 3 failure cases.
  - We use one unified model for all failure cases, while NOMS2020 paper use different models to predict network and network failures separately.
  - We also add Decision Tree (DT) and XGBoost (XGB) to the comparative analysis and point out that XGBoost has a comparable performance as Random Forest.


Thanks
UT-NakaoLab-AI Team
2020-10-15