Opprentice: Towards Practical and Automatic Anomaly Detection Through Machine Learning

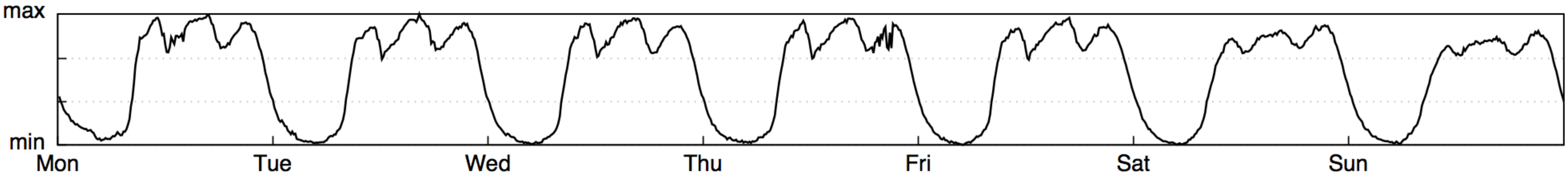
Dapeng Liu, Youjian Zhao, Haowen Xu, Yongqian Sun, Dan Pei, Jiao Luo, Xiaowei Jing, Mei Feng

Tsinghua

Baidu 百度

PetroChina

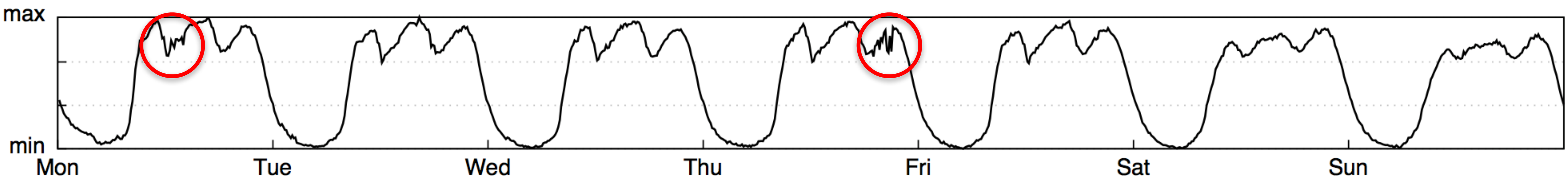
KPIs and Anomaly Detection



Page views (PV) of Baidu

KPIs (Key Performance Indicators): A set of performance measures that evaluate the service quality

KPIs and Anomaly Detection

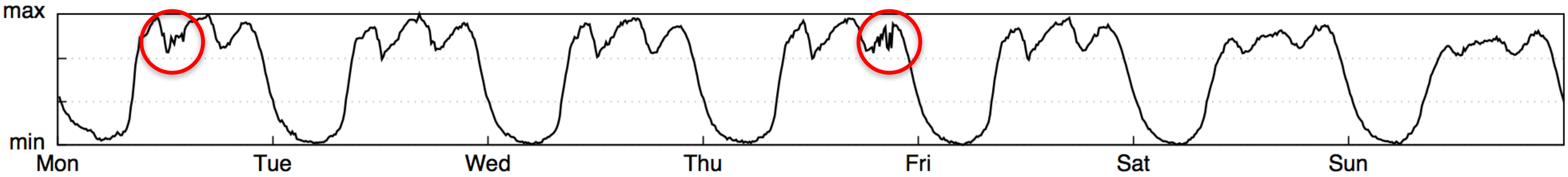


Page views (PV) of Baidu

KPIs (Key Performance Indicators): A set of performance measures that evaluate the service quality

KPI anomalous (unexpected) behaviors → Potential failures, bugs, attacks...

KPIs and Anomaly Detection



Page views (PV) of Baidu

KPIs (Key Performance Indicators): A set of performance measures that evaluate the service quality

KPI anomalous (unexpected) behaviors → Potential failures, bugs, attacks...

Anomaly detection matters: Find anomalous behaviors of the KPI curve

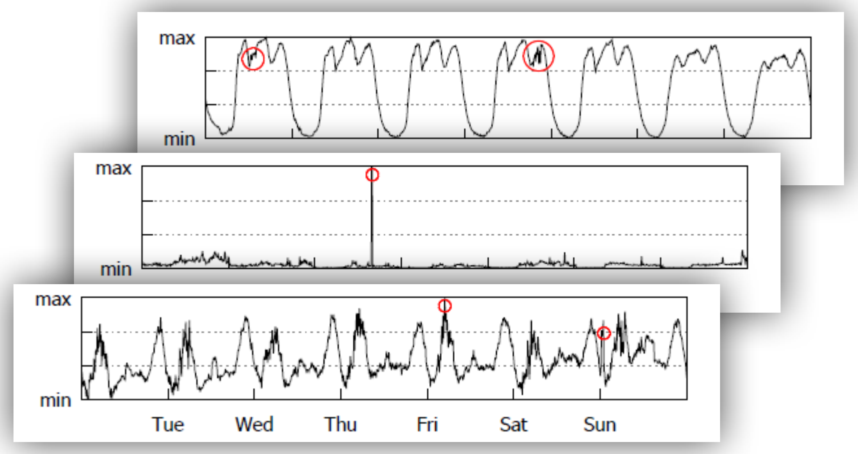
→ Diagnose and fix it

→ Avoid further influences and revenue losses

How to Build an Anomaly Detection System

Domain experts (Operators)
- Responsible for the KPIs
- Knowing the KPI behaviors well

Developers
- Building the detection system
- Knowing several anomaly detectors



Simple threshold

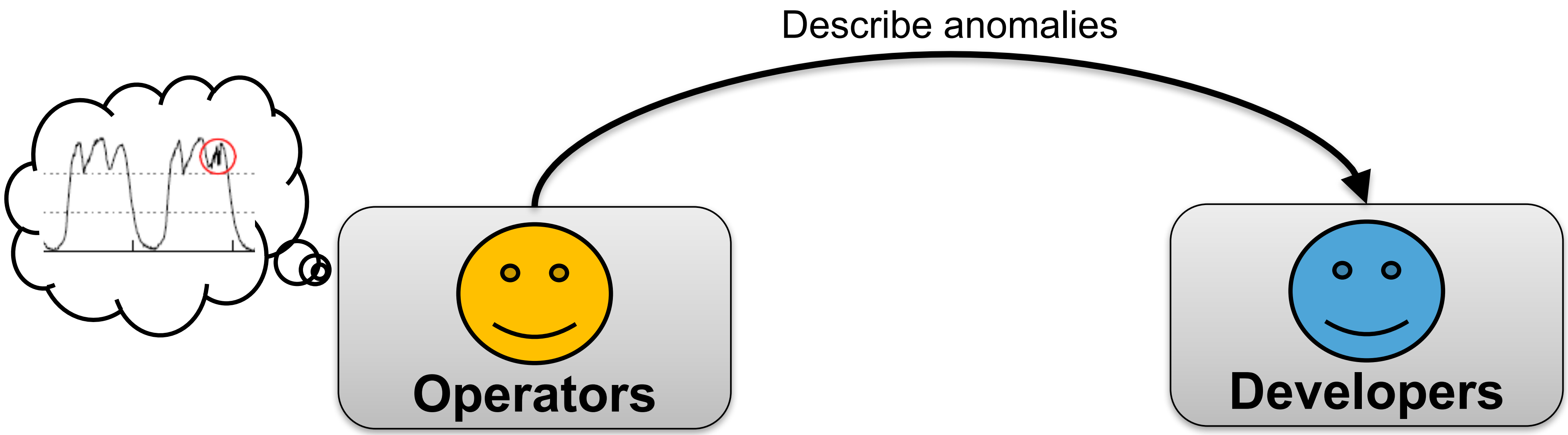
Historical Average

Wavelet

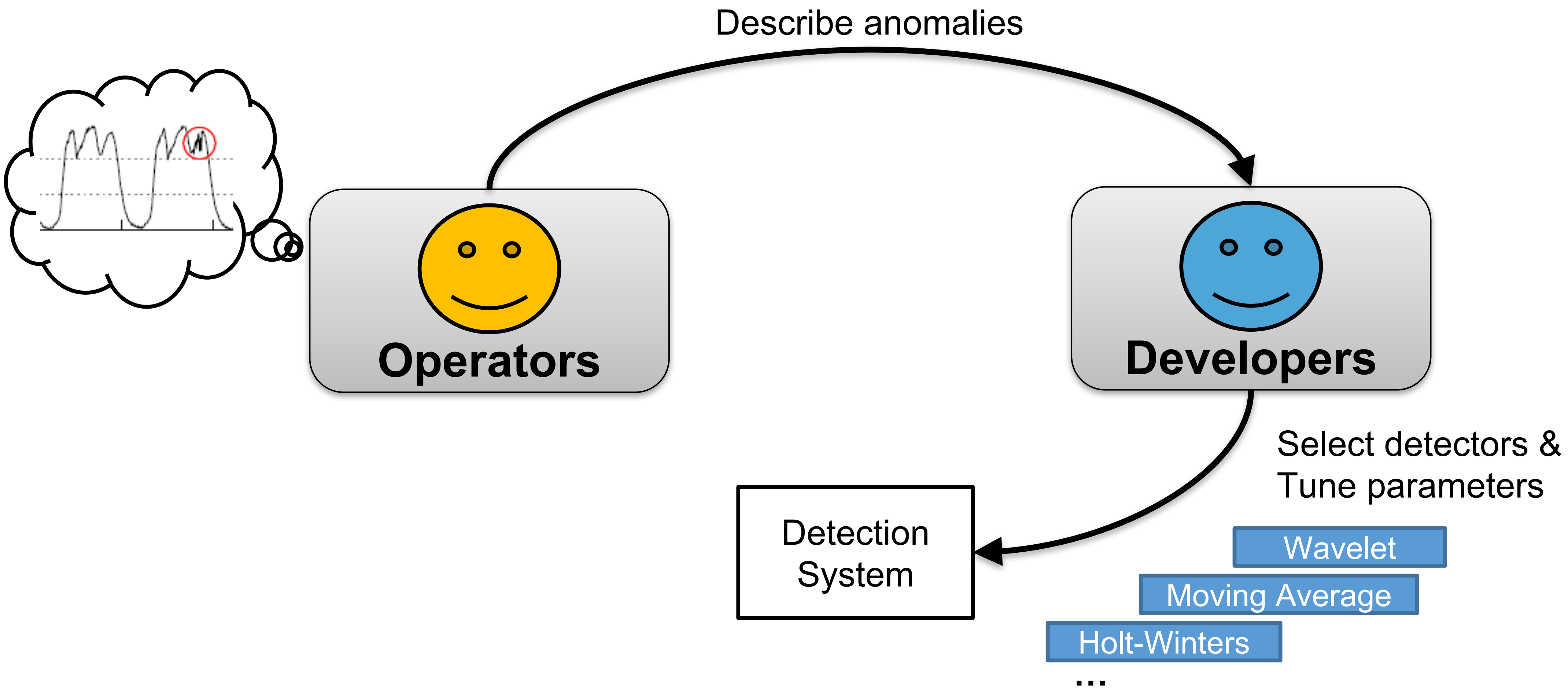
Holt-Winters

…

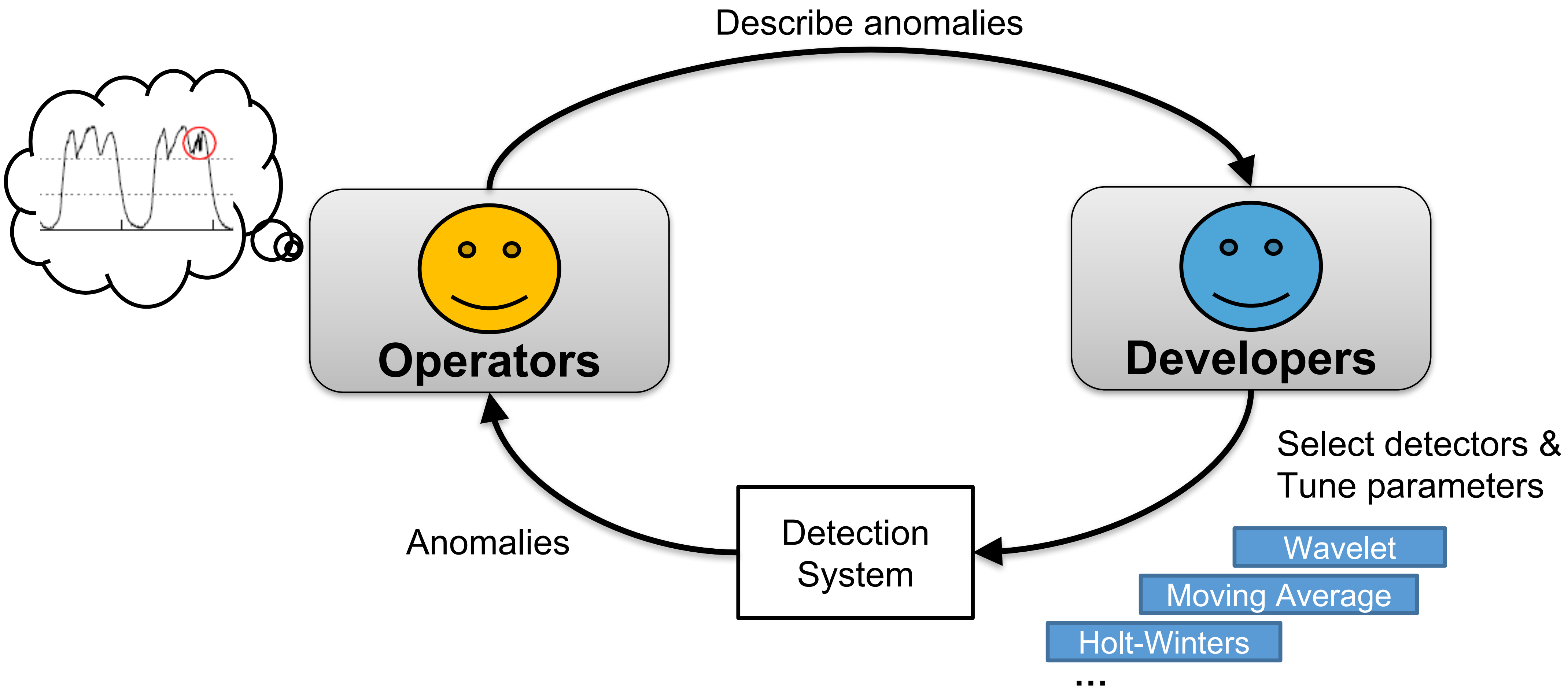
In practice, it is more complex



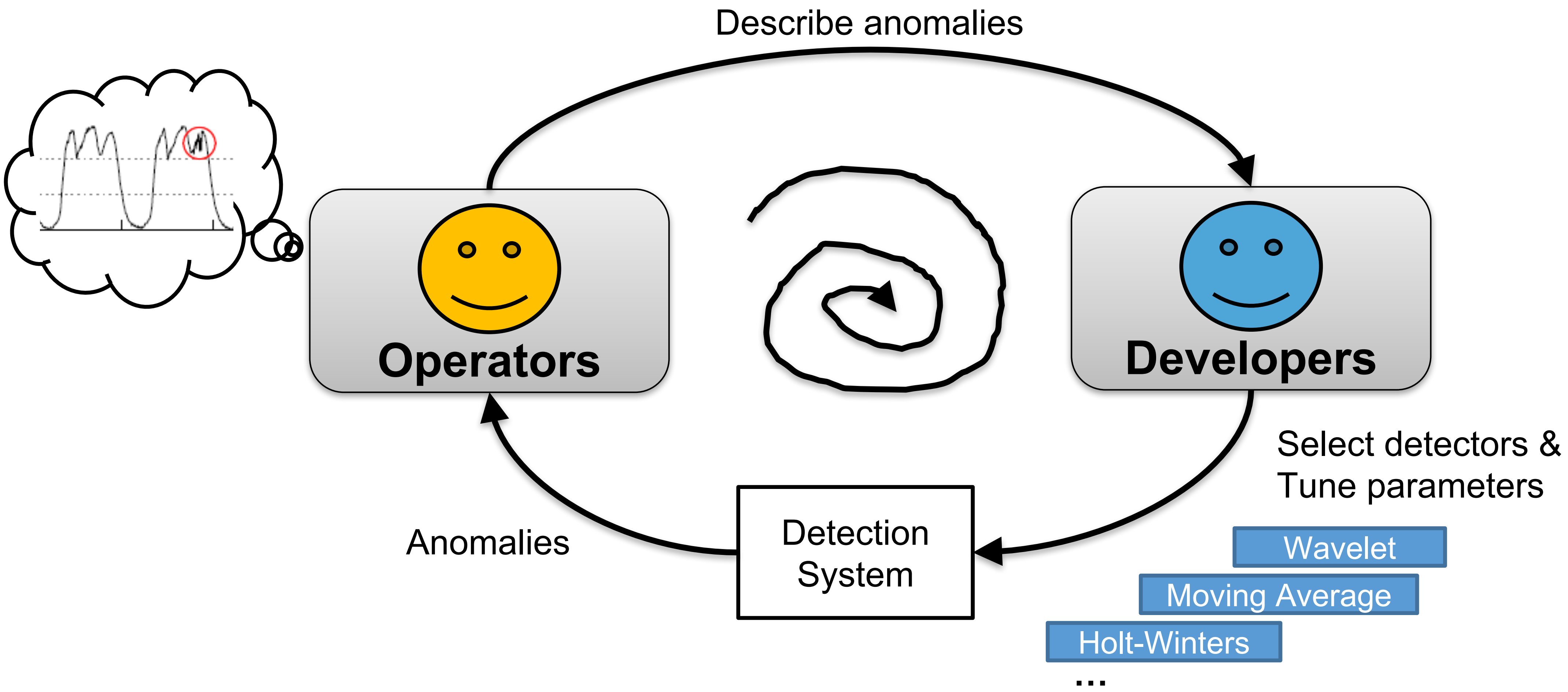
In practice, it is more complex



In practice, it is more complex



In practice, it is more complex



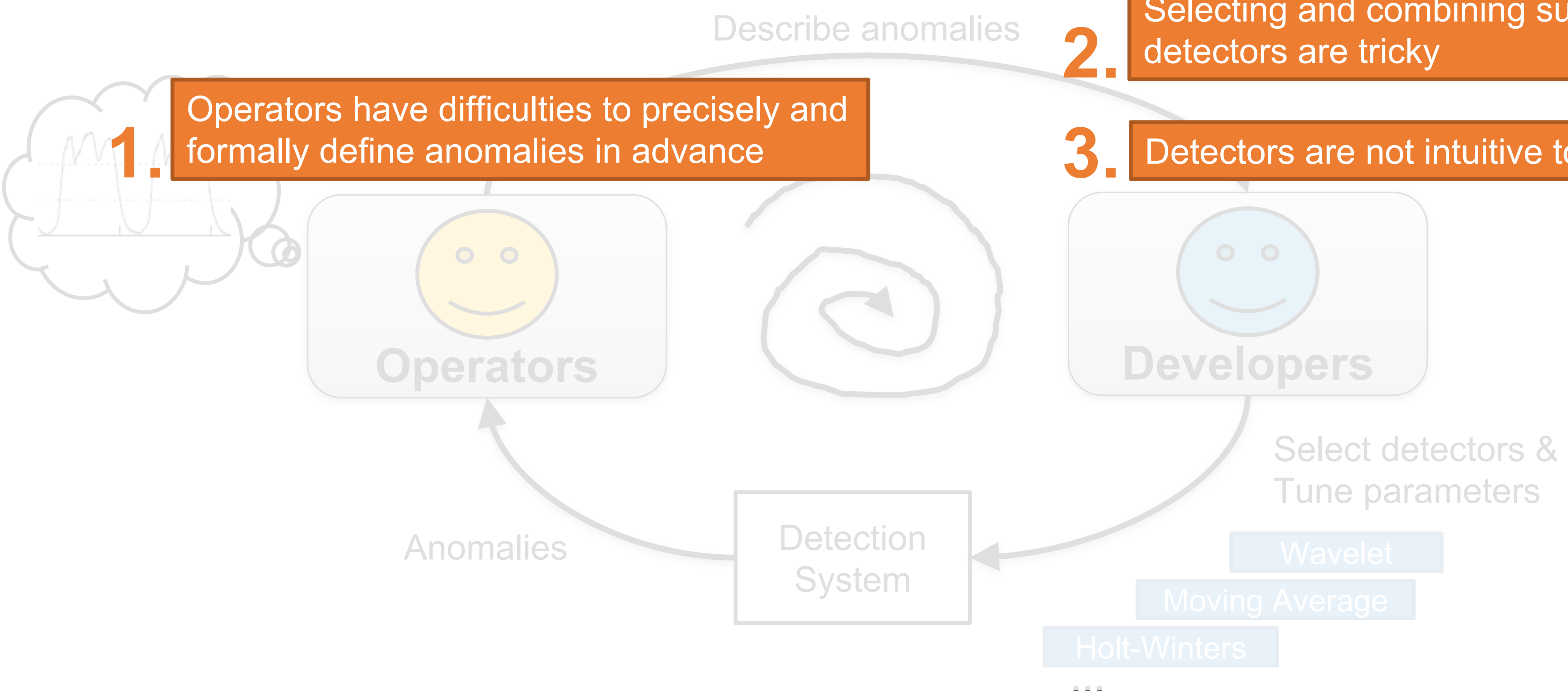
How to Build an Anomaly Detection System

Challenges

1. Operators have difficulties to precisely and formally define anomalies in advance

2. Selecting and combining suitable detectors are tricky

3. Detectors are not intuitive to tune

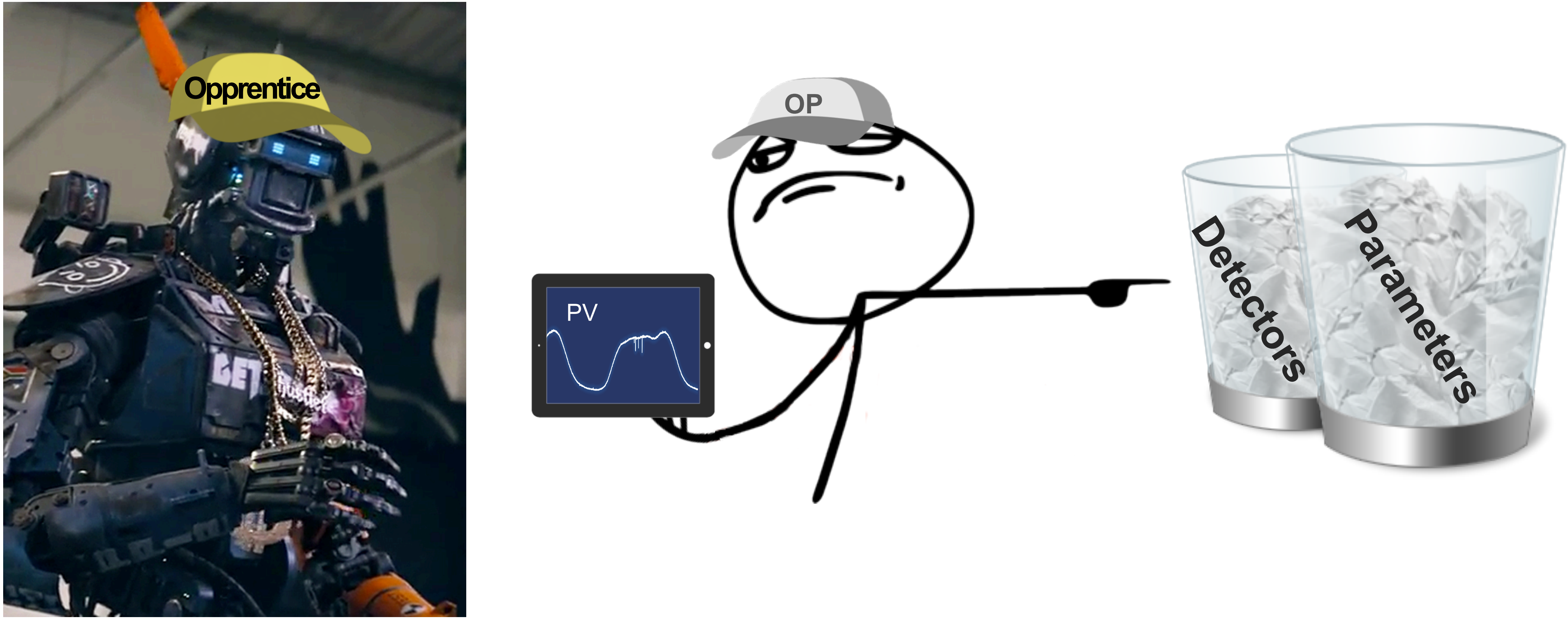



CHAPPiE
CHAPPIE
DADDY

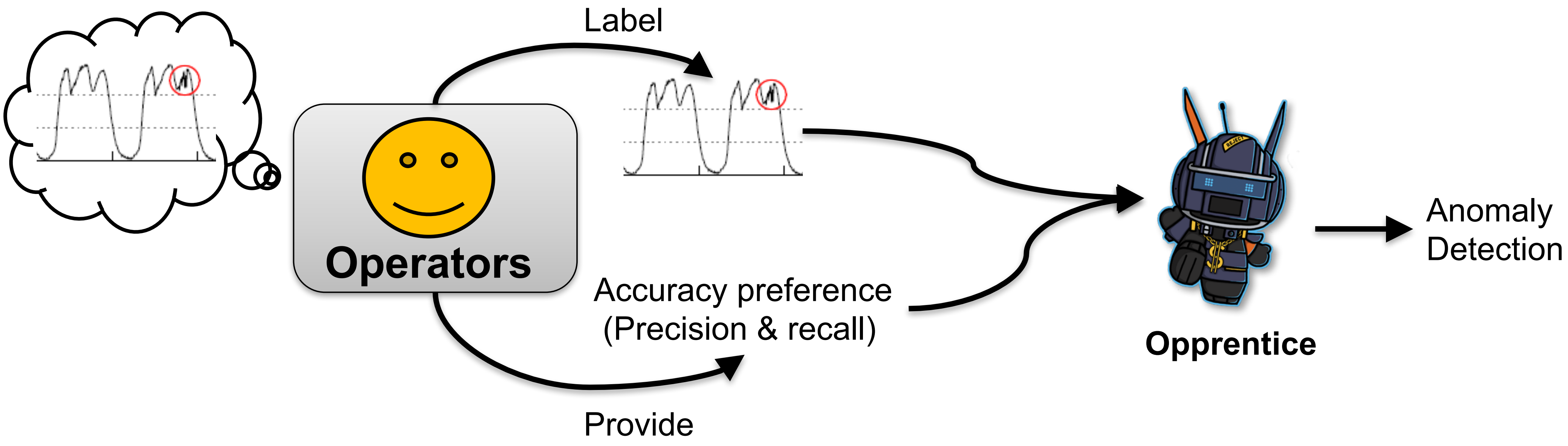
Opprentice

(Operators' apprentice)

A More Natural Way

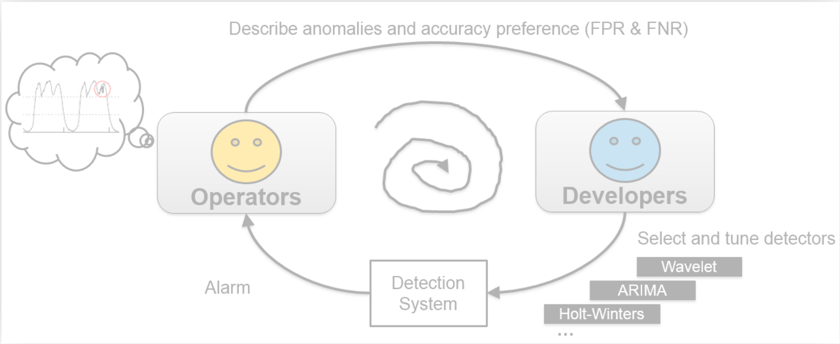


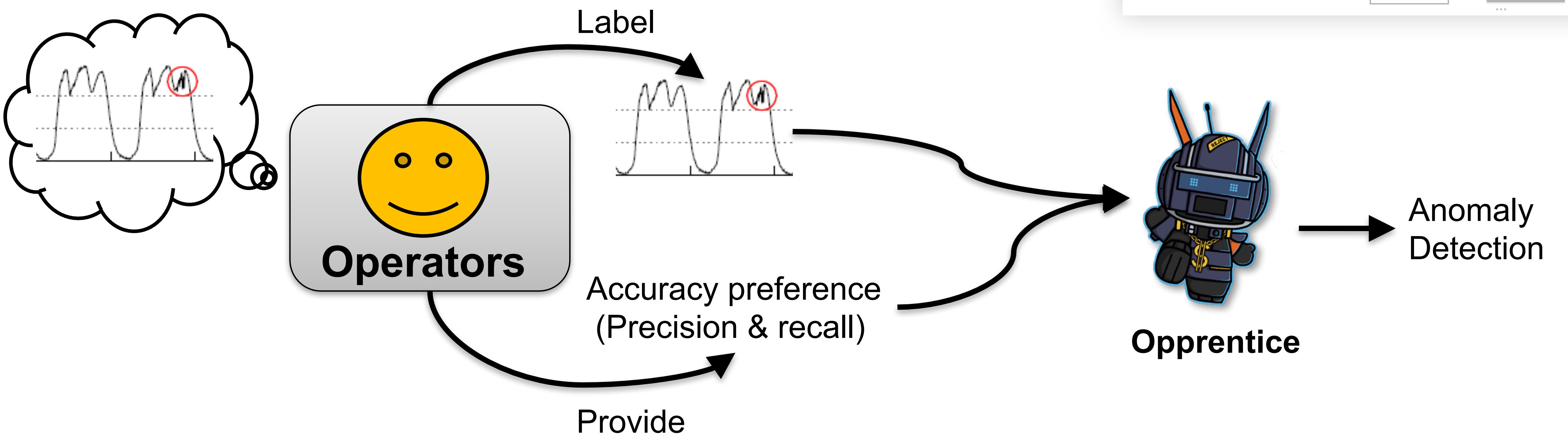
Design Goal


Label
Operators
Accuracy preference
(Precision & recall)
Provide
Opprentice
Anomaly
Detection

Design Goal

VS.





Outline

- Background and Motivation
- **Key Ideas**
- Results
- Conclusion

Detector model:

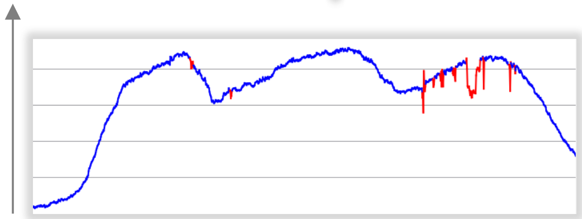
$$\text{data point} \xrightarrow{\text{a detector with parameters } \{p\}} \text{severity} \xrightarrow{\text{sThld}} \{1, 0\}$$

Detector model:

$$\text{data point} \xrightarrow{\text{a detector with parameters } \{p\}} \text{severity} \xrightarrow{\text{sThld}} \{1, 0\}$$

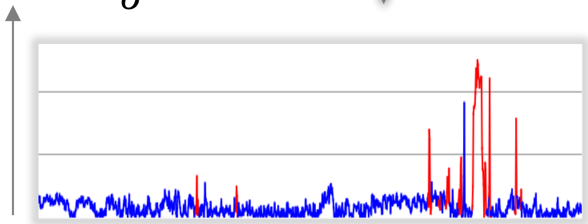
For example

$value$



Historical Average

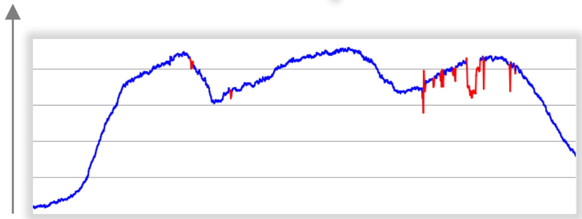
$$\text{severity} = \frac{|value - \mu|}{\sigma}$$



A generalized model of anomaly detection algorithm based on time series algorithm

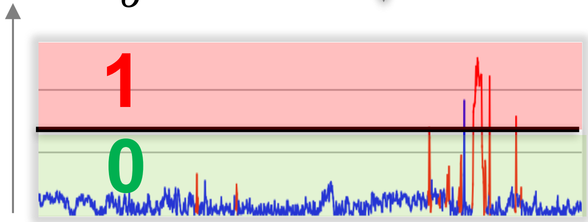
$$\text{data point} \xrightarrow{\text{a detector with parameters } \{p\}} \text{severity} \xrightarrow{\text{sThld}} \{1,0\}$$

$value$



Historical Average

$$\text{severity} = \frac{|value - \mu|}{\sigma}$$



sThld

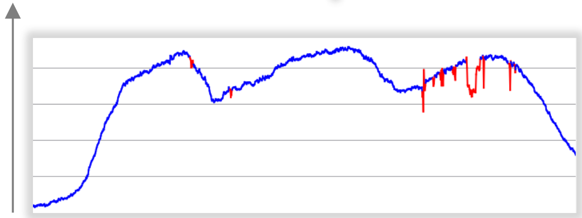
Key Ideas

Detector model:

$$\text{data point} \xrightarrow{\text{a detector with parameters } \{p\}} \text{severity} \xrightarrow{\text{sThld}} \{1, 0\}$$

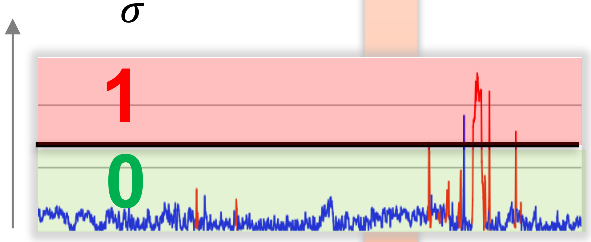
For example

$value$



Historical Average

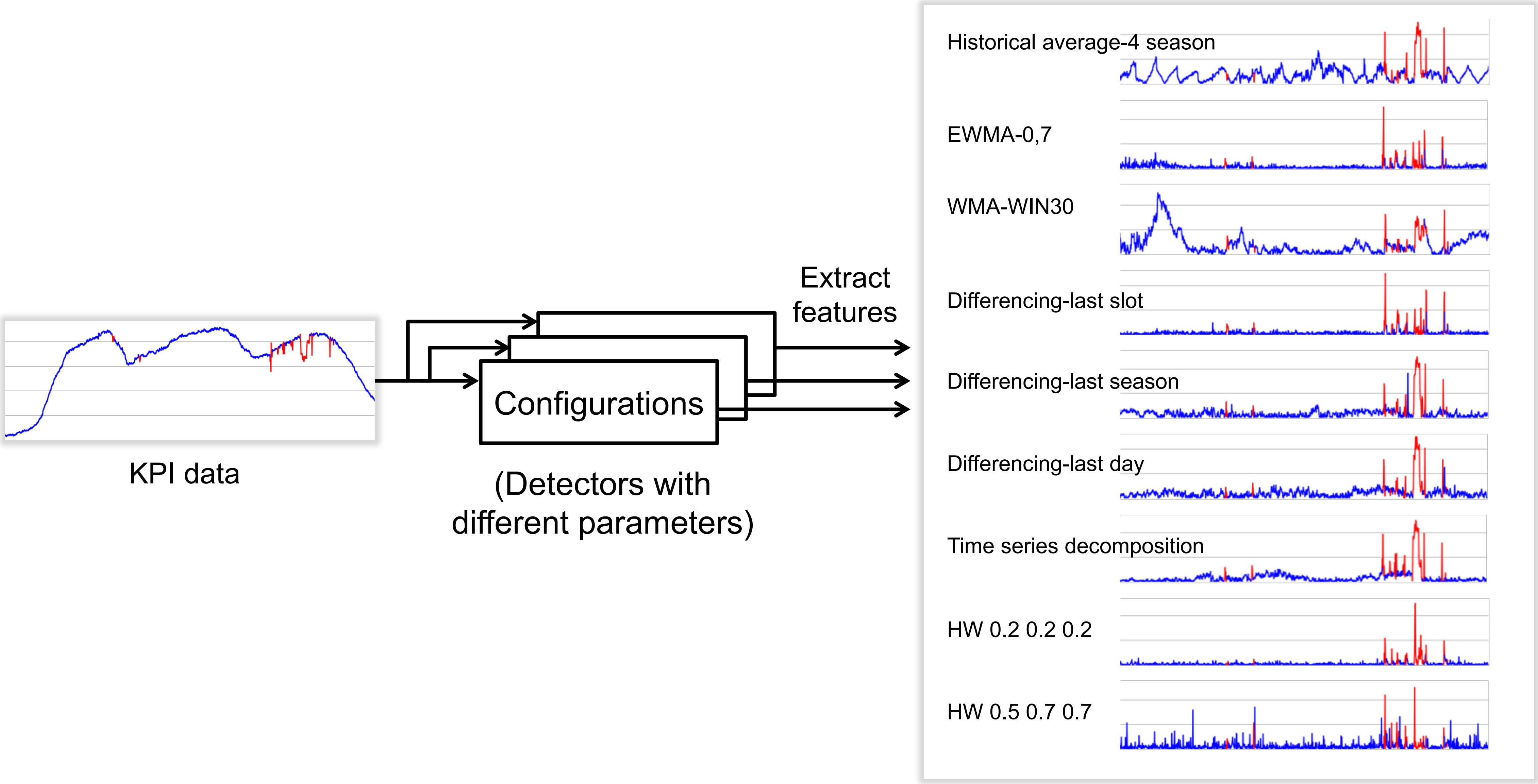
$$\text{severity} = \frac{|value - \mu|}{\sigma}$$



sThld

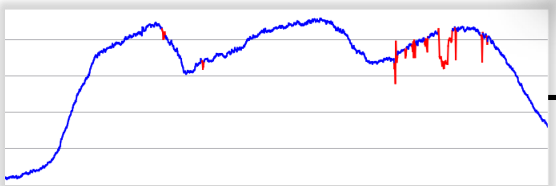
Anomaly feature

Key Ideas

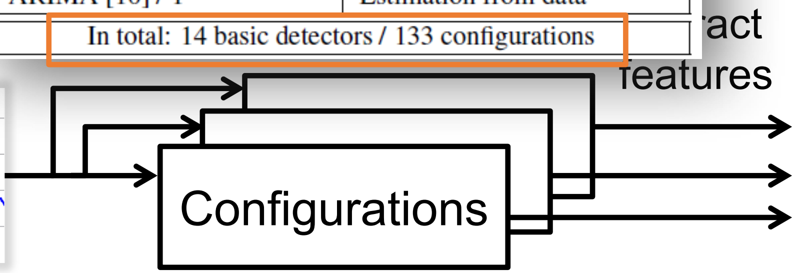

KPI data
Configurations
(Detectors with
different parameters)
Extract
features
Historical average-4 season
EWMA-0,7
WMA-WIN30
Differencing-last slot
Differencing-last season
Differencing-last day
Time series decomposition
HW 0.2 0.2 0.2
HW 0.5 0.7 0.7

Key Ideas

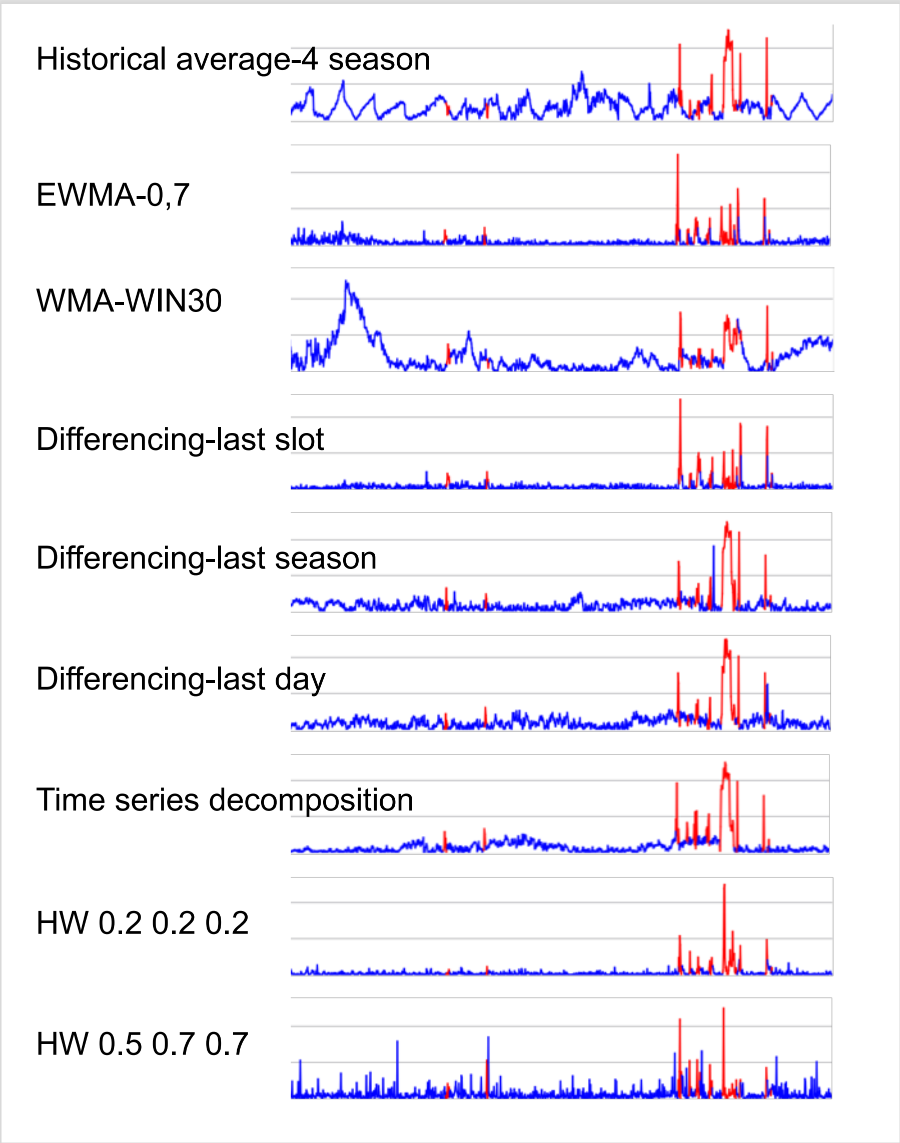
Detector / # of configurations	Sampled parameters
Simple threshold [24] / 1	none
Diff / 3	last-slot, last-day, last-week
Simple MA [4] / 5	win = 10, 20, 30, 40, 50 points
Weighted MA [11] / 5	
MA of diff / 5	
EWMA [11] / 5	$\alpha = 0.1, 0.3, 0.5, 0.7, 0.9$
TSD [1] / 5	win = 1, 2, 3, 4, 5 week(s)
TSD MAD / 5	
Historical average [5] / 5	
Historical MAD / 5	
Holt-Winters [6] / $4^3 = 64$	$\alpha, \beta, \gamma = 0.2, 0.4, 0.6, 0.8$
SVD [7] / $5 \times 3 = 15$	row =10, 20, 30, 40, 50 points, column =3, 5, 7
Wavelet [12] / $3 \times 3 = 9$	win = 3, 5 ,7 days, freq = low, mid, high
ARIMA [10] / 1	Estimation from data
In total: 14 basic detectors / 133 configurations	



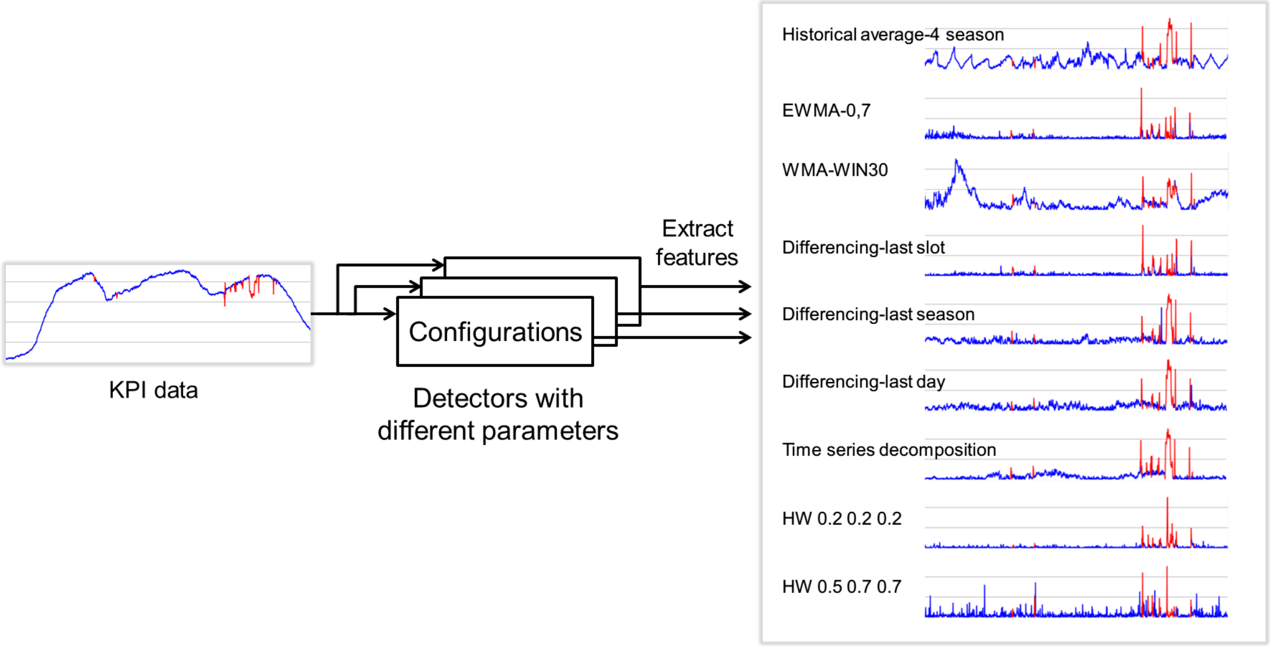
KPI data



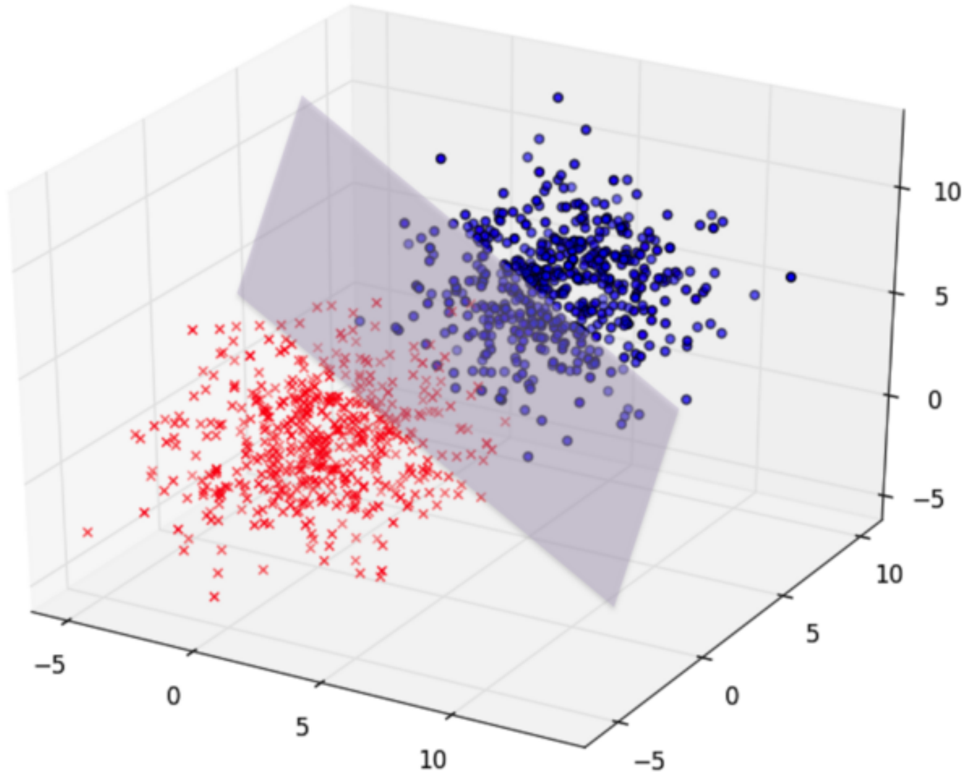
(Detectors with different parameters)



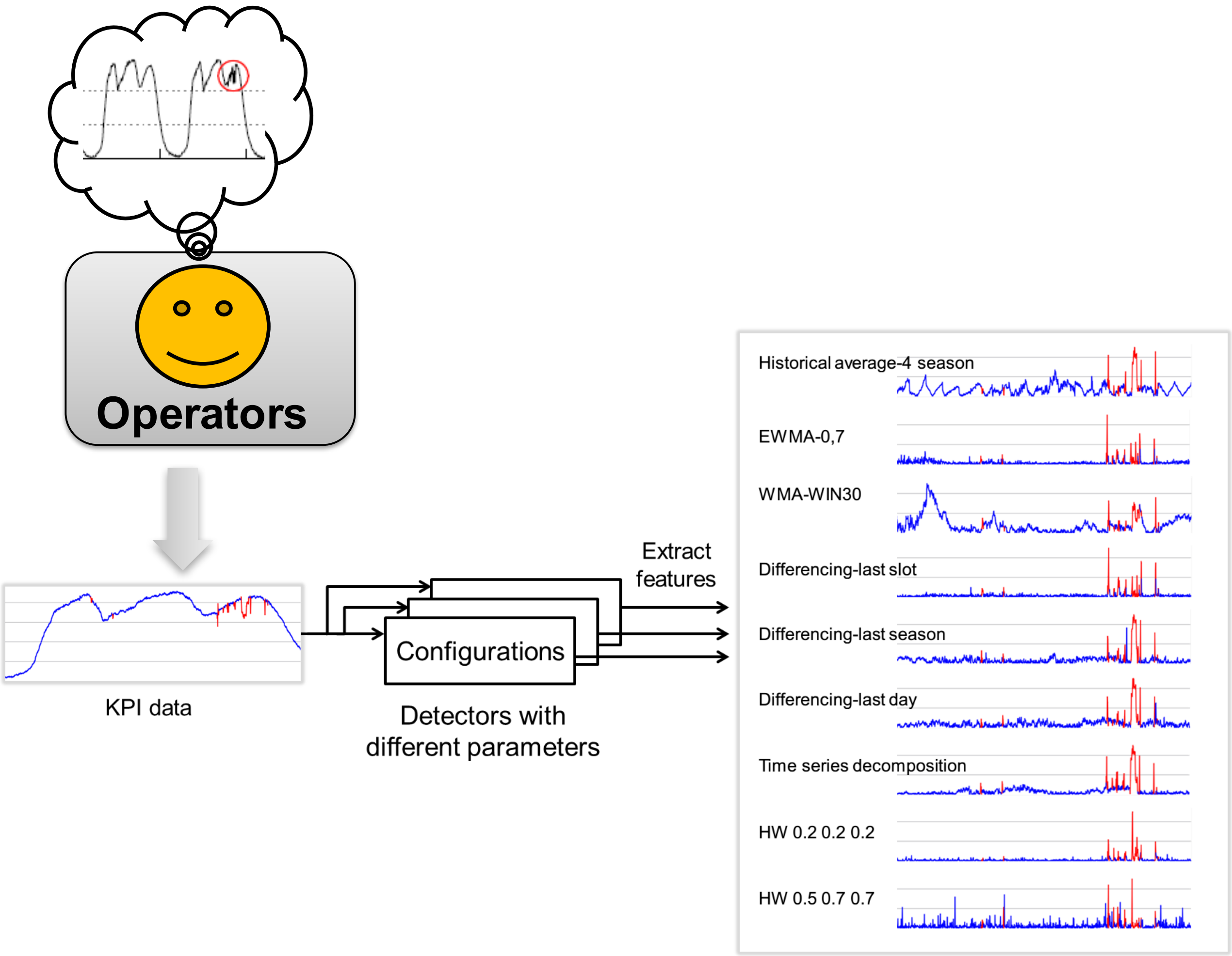
Key Ideas



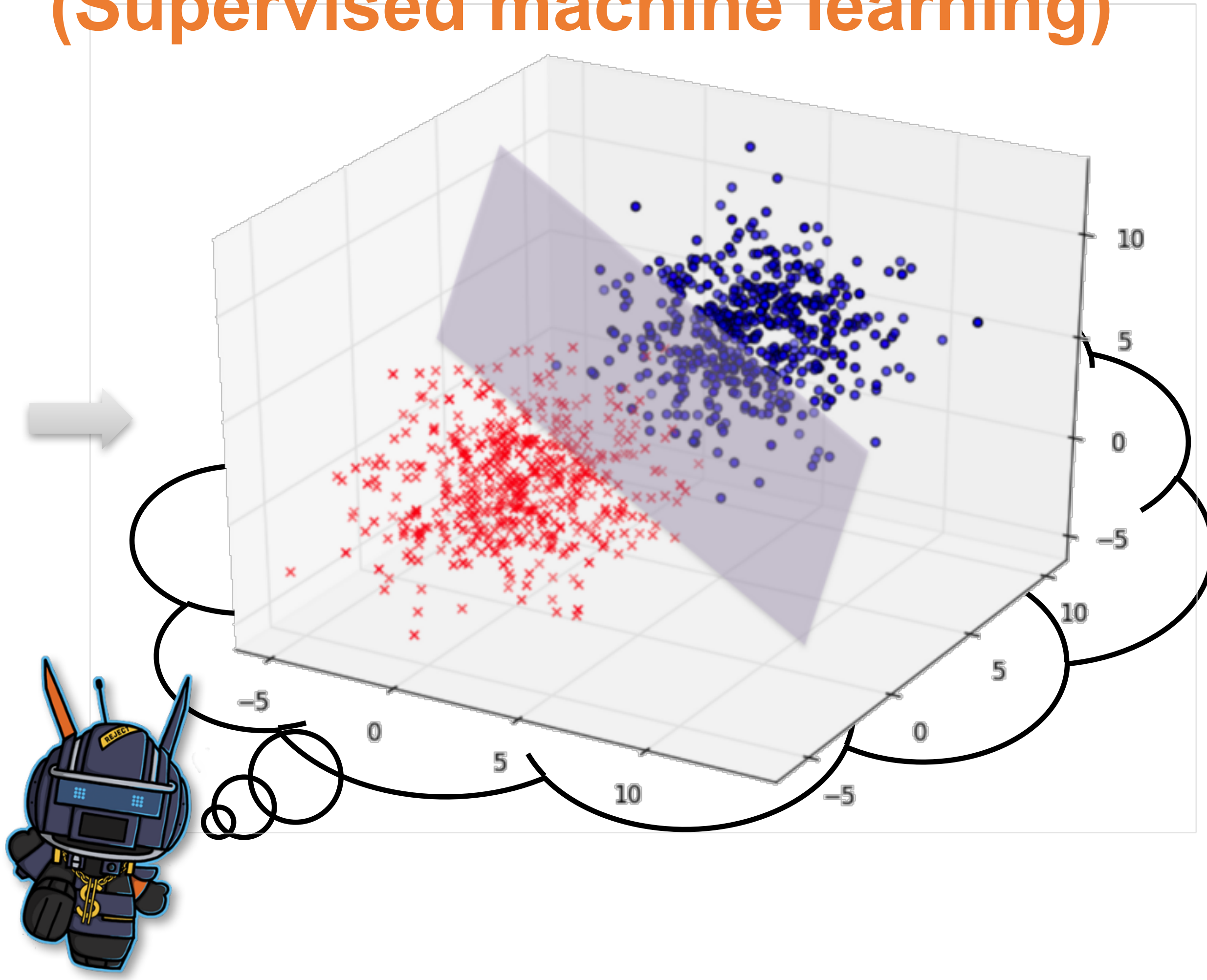
Classification in the feature space (Supervised machine learning)



Key Ideas

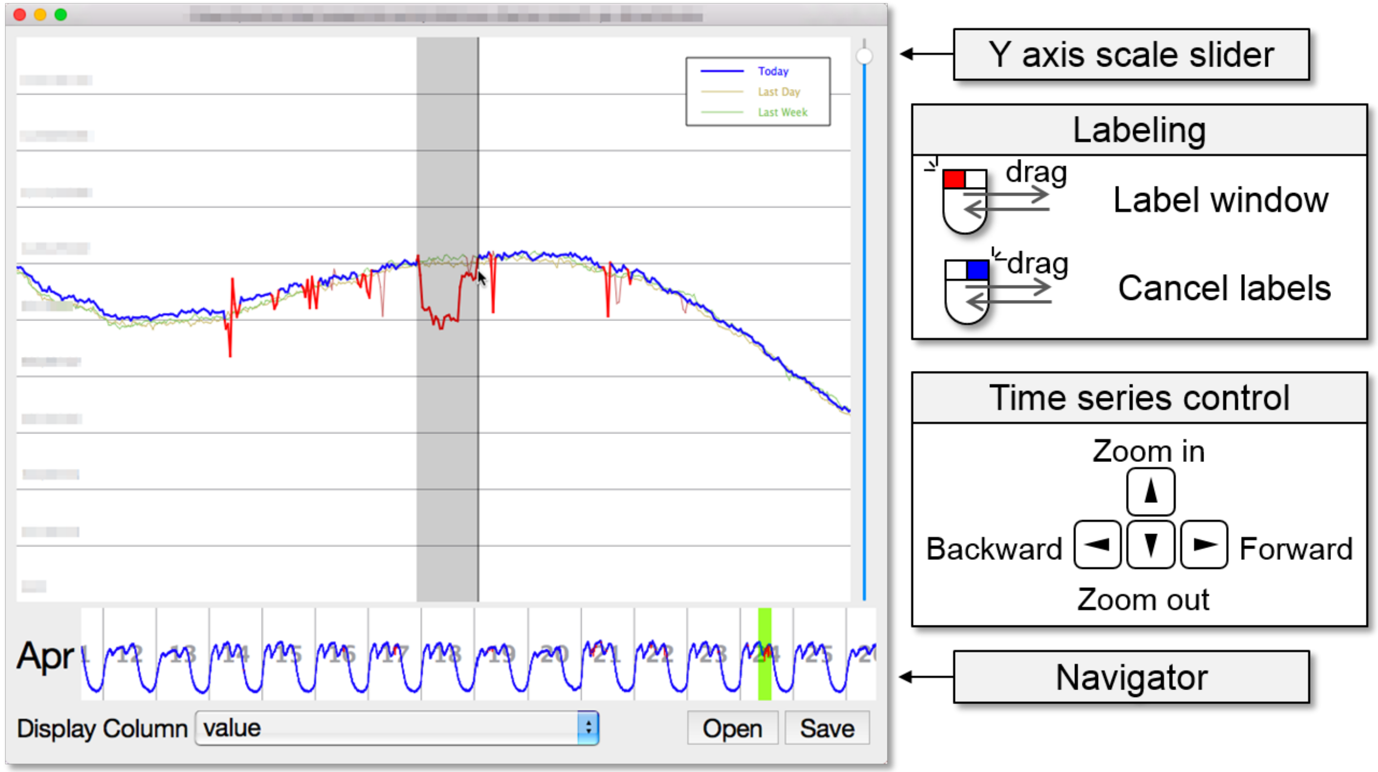


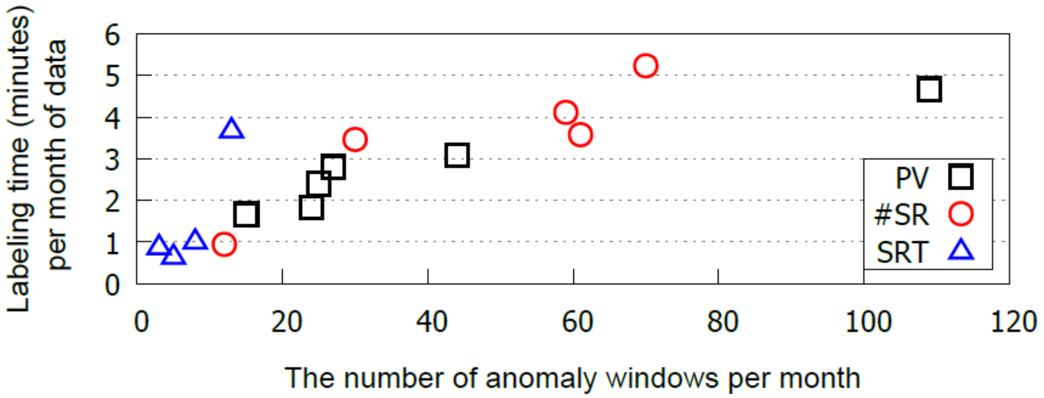
Classification in the feature space (Supervised machine learning)



Address Challenges of Designing Opprentice

- Labeling overhead
  - Solution: an effective labeling tool





Address Challenges of Designing Opprentice

- Labeling overhead
  - Solution: an effective labeling tool
- Incomplete anomaly types in the historical data
  - Solution: incremental re-training with new data

Address Challenges of Designing Opprentice

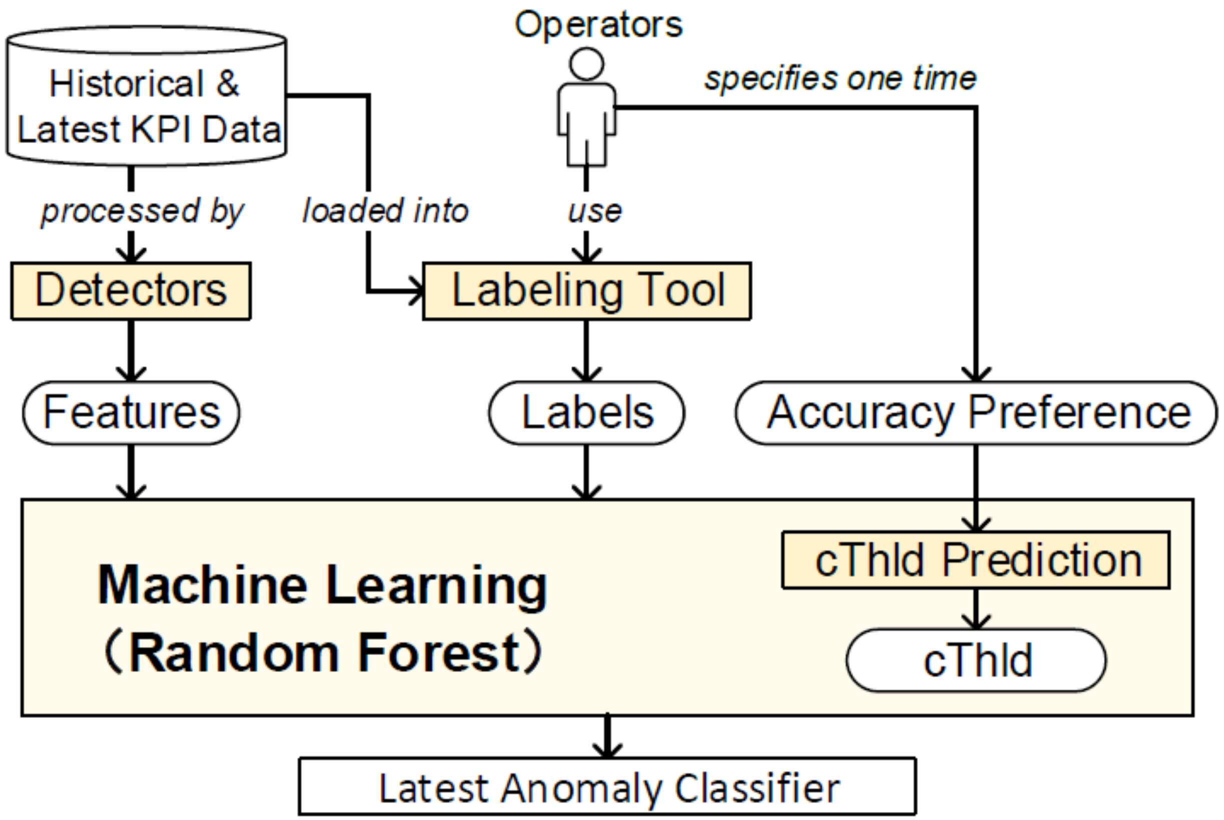
- Labeling overhead
  - Solution: an effective labeling tool
- Incomplete anomaly types in the historical data
  - Solution: incremental re-training with new data
- Class imbalance problem
  - Solution: adjusting classification threshold (cThld) based on the preference

Address Challenges of Designing Opprentice

- Labeling overhead
  - Solution: an effective labeling tool
- Incomplete anomaly types in the historical data
  - Solution: incremental re-training with new data
- Class imbalance problem
  - Solution: adjusting classification threshold (cThld) based on the preference
- Irrelevant and redundant features
  - Solution: random forests

Design Overview

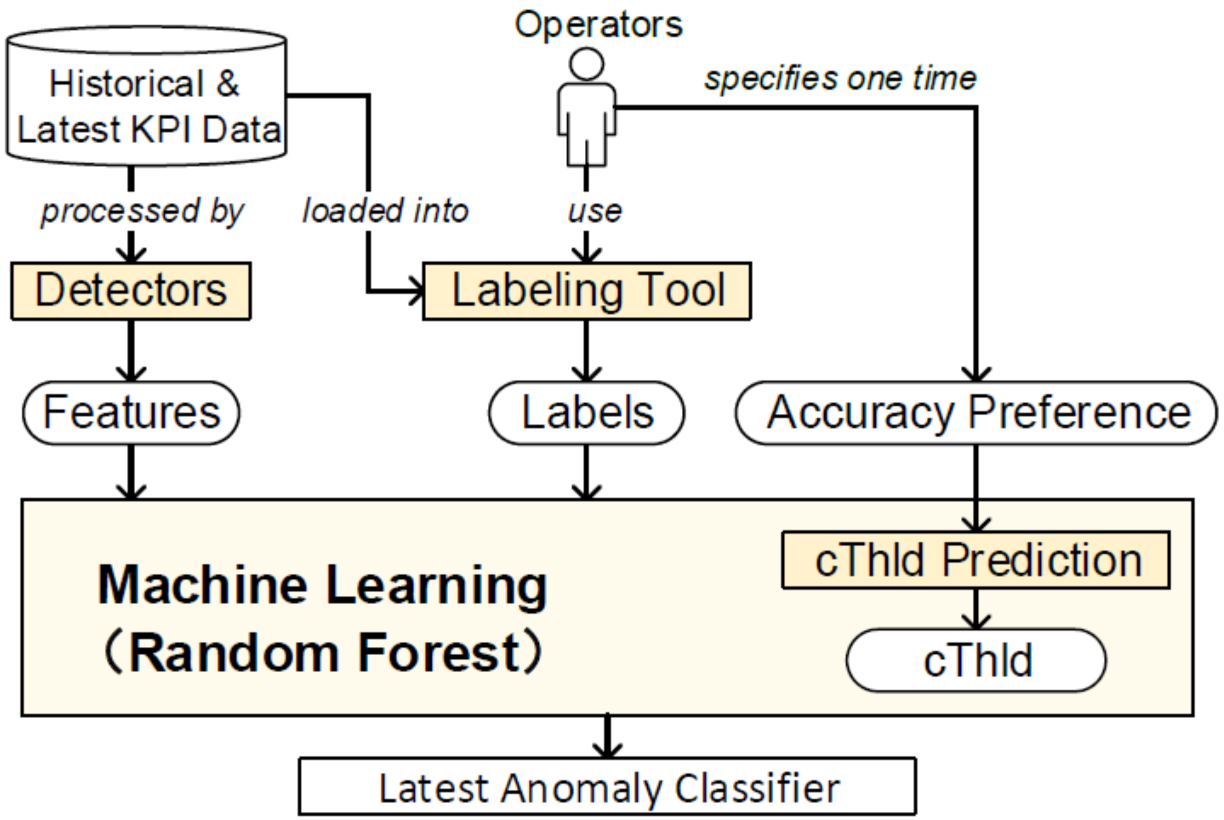
Training a classifier



See the paper for full details

Design Overview

Training a classifier



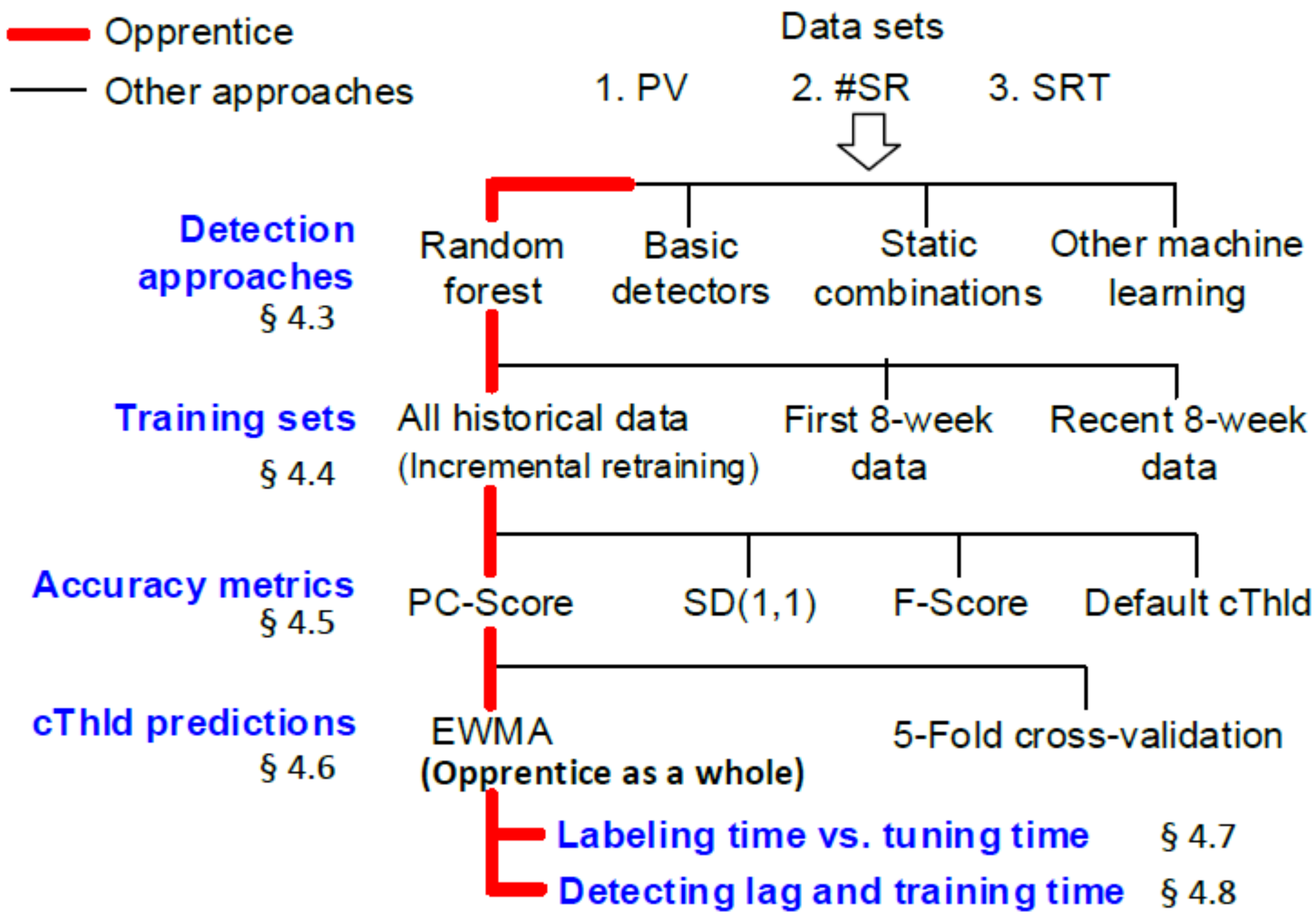
See the paper for full details

Detecting anomalies

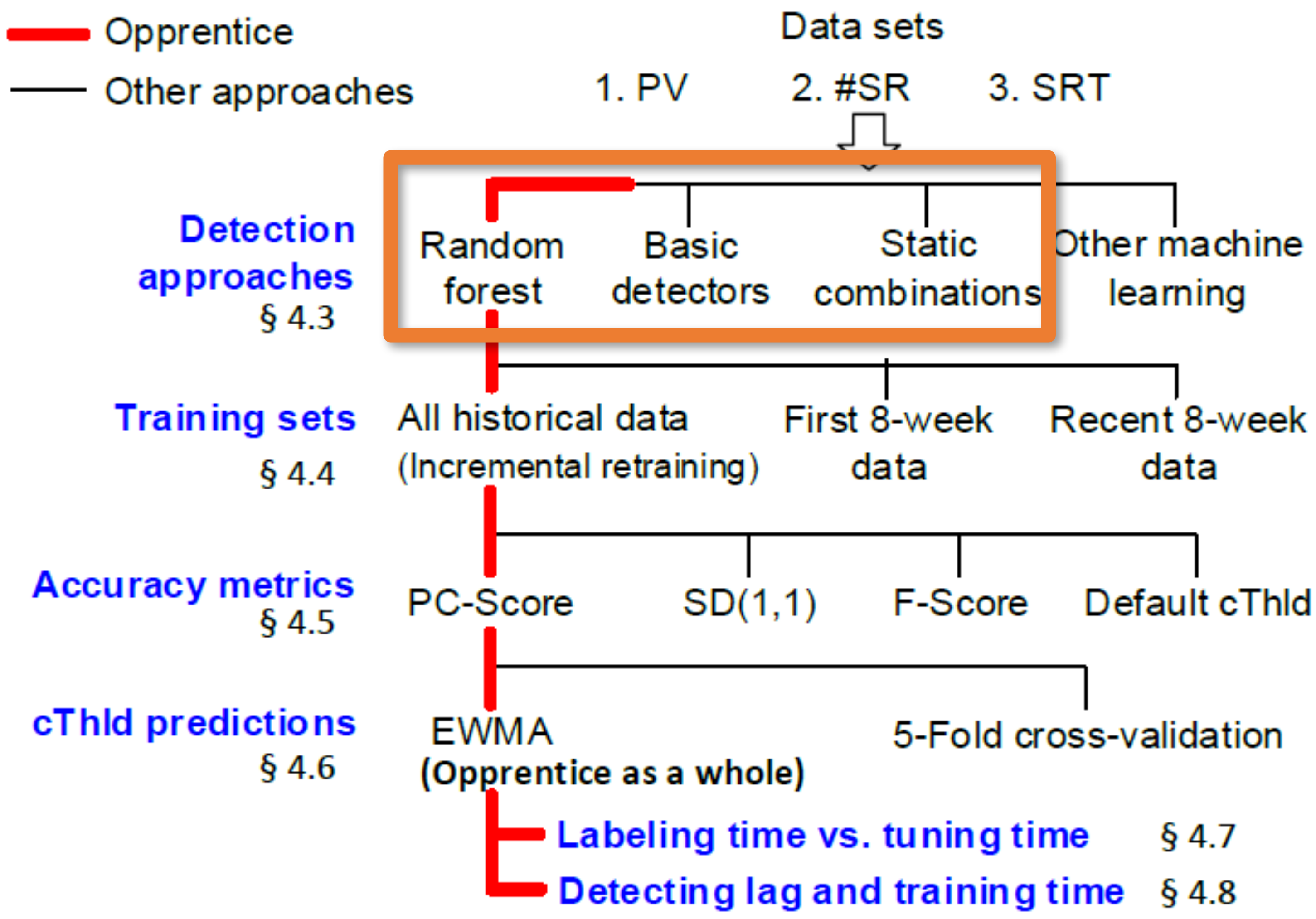
Outline

- Background and Motivation
- Key Ideas
- **Results**
- Conclusion

Evaluation

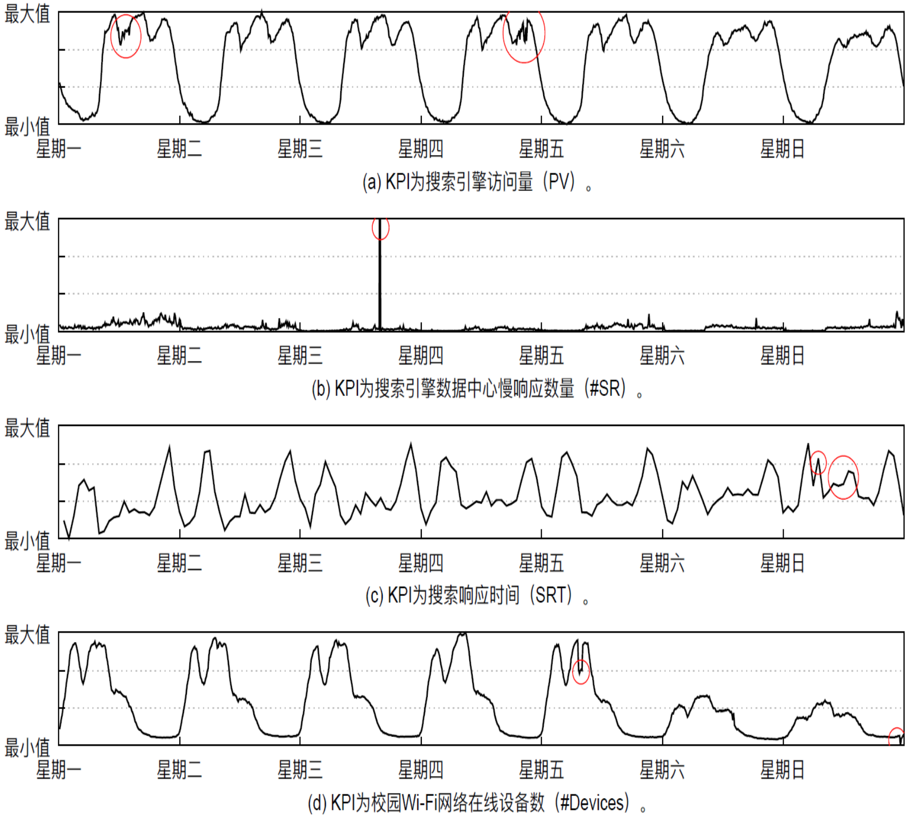

Opprentice
Other approaches
Data sets
1. PV
2. #SR
3. SRT
Detection approaches
§ 4.3
Random forest
Basic detectors
Static combinations
Other machine learning
Training sets
§ 4.4
All historical data (Incremental retraining)
First 8-week data
Recent 8-week data
Accuracy metrics
§ 4.5
PC-Score
SD(1,1)
F-Score
Default cThld
cThld predictions
§ 4.6
EWMA (Opprentice as a whole)
5-Fold cross-validation
Labeling time vs. tuning time
§ 4.7
Detecting lag and training time
§ 4.8

Evaluation


Opprentice
Other approaches
Data sets
1. PV
2. #SR
3. SRT
Detection approaches
§ 4.3
Random forest
Basic detectors
Static combinations
Other machine learning
Training sets
§ 4.4
All historical data (Incremental retraining)
First 8-week data
Recent 8-week data
Accuracy metrics
§ 4.5
PC-Score
SD(1,1)
F-Score
Default cThld
cThld predictions
§ 4.6
EWMA (Opprentice as a whole)
5-Fold cross-validation
Labeling time vs. tuning time
§ 4.7
Detecting lag and training time
§ 4.8

Evaluation

- 四种真实KPI数据



(a) KPI为搜索引擎访问量（PV）。

(b) KPI为搜索引擎数据中心慢响应数量（#SR）。

(c) KPI为搜索响应时间（SRT）。

(d) KPI为校园Wi-Fi网络在线设备数（#Devices）。

Search PV (25 weeks) ）

#slow queries (19 weeks)

Search Response Time (16 weeks)

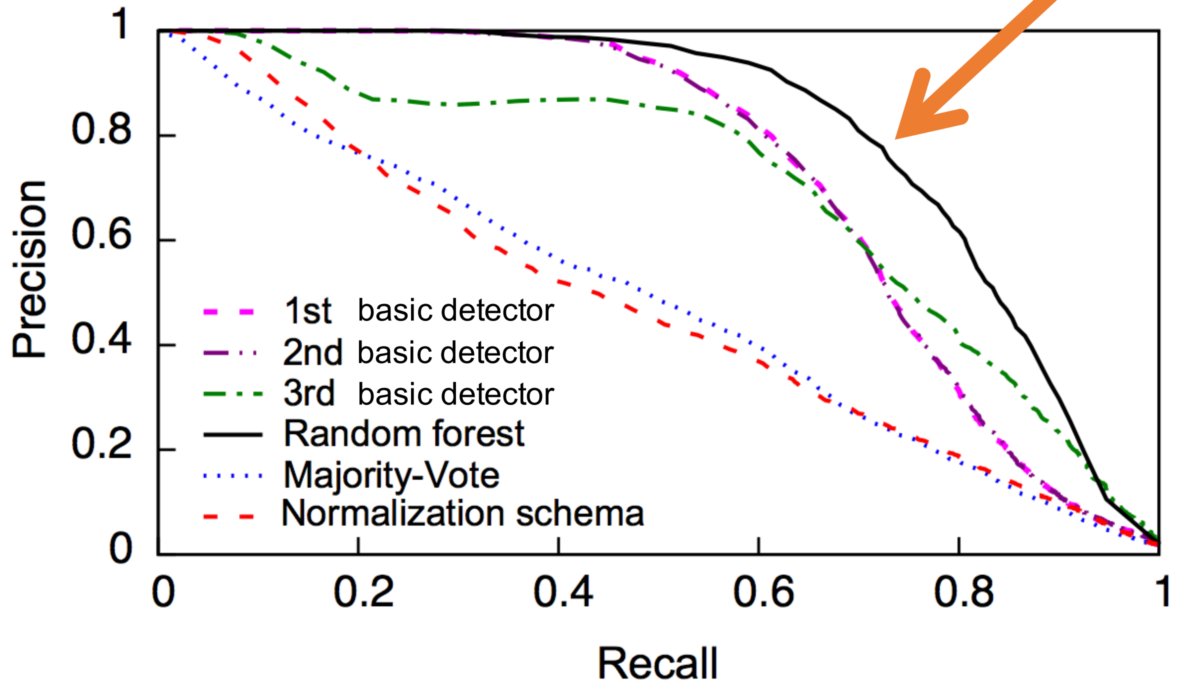
#online devices (15 weeks)

Baidu

Tsinghua Enterprise WiFi

Random forests vs. Basic Detectors and Static Combinations

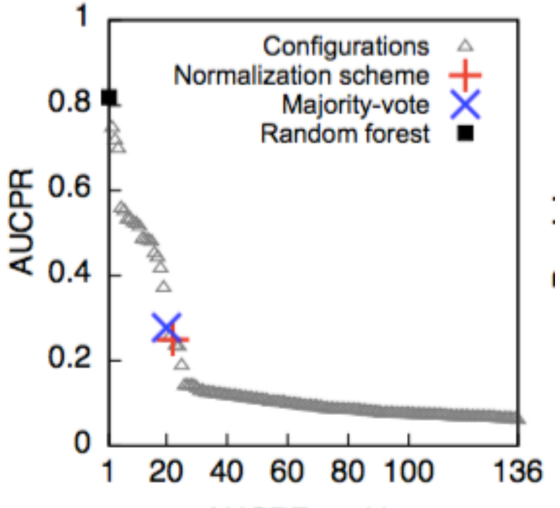
Random forest

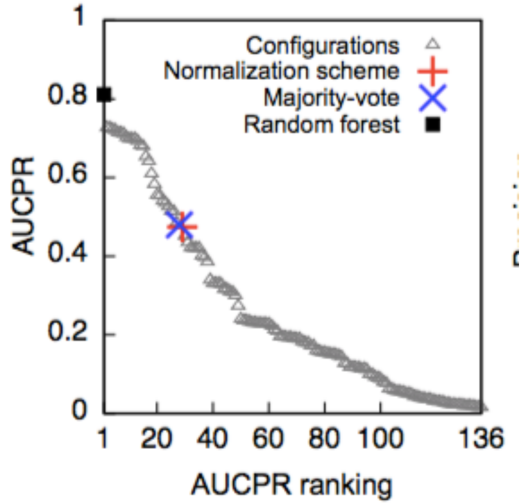


Evaluation

- Compared with all existing detectors (Four KPIs)

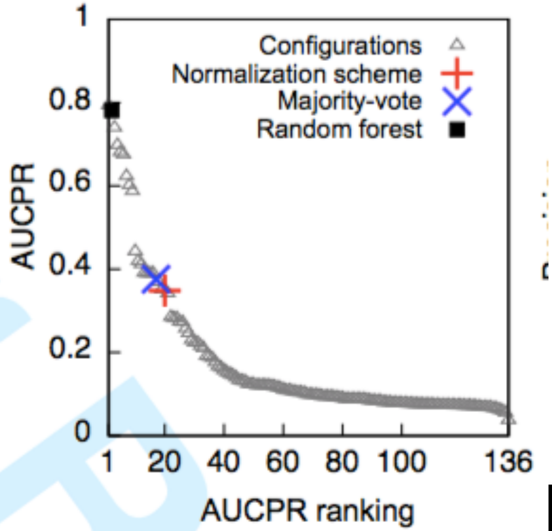
first

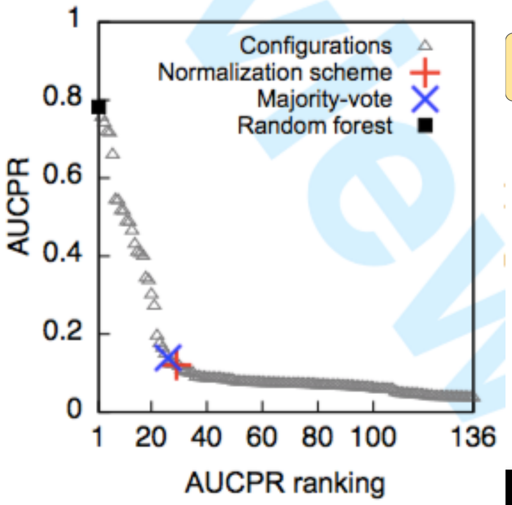




first

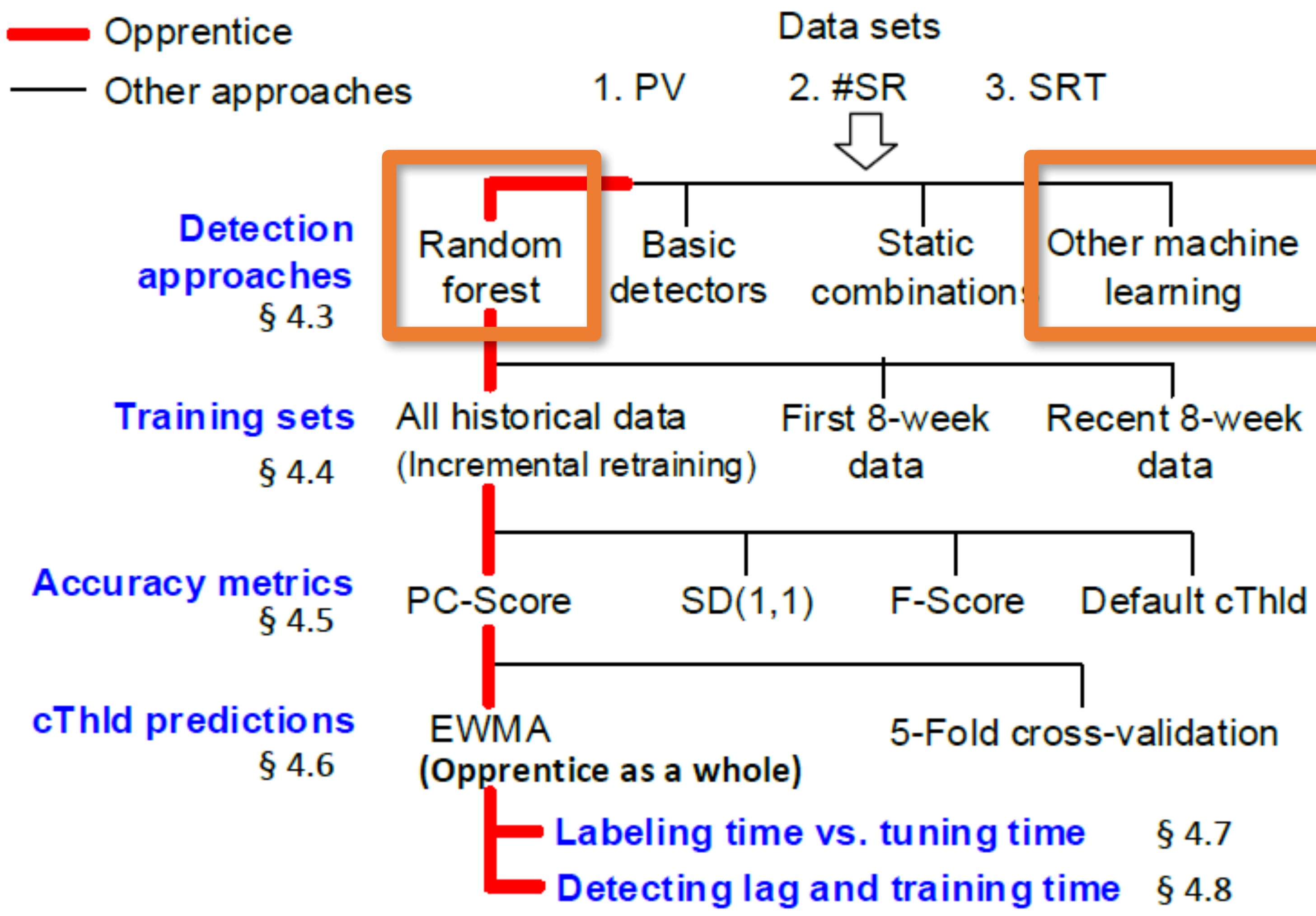
second





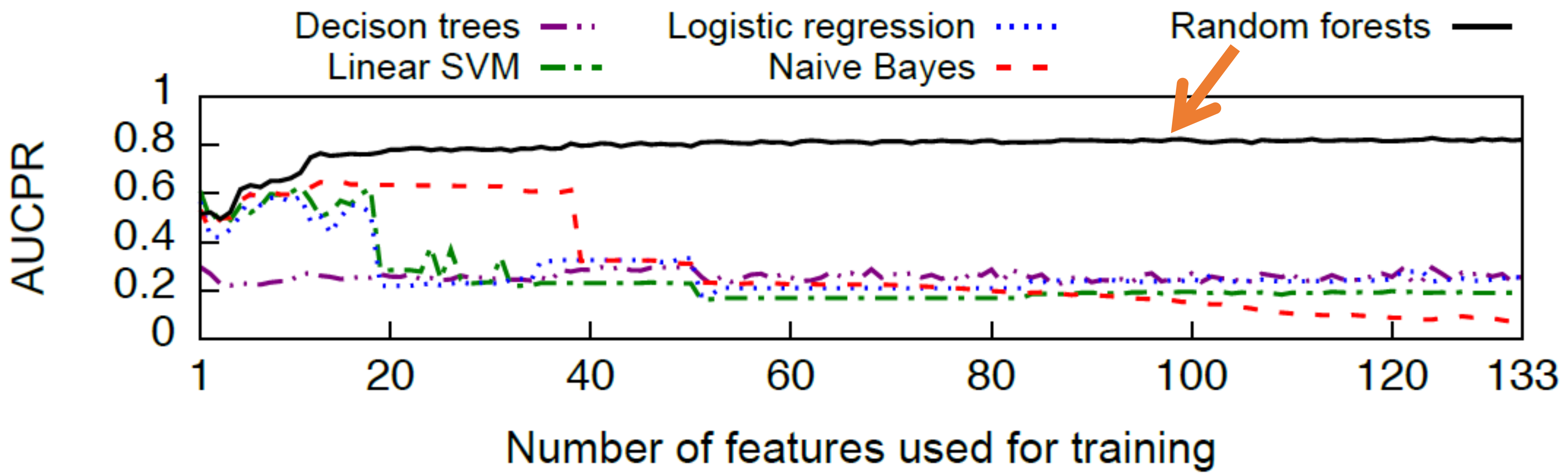
first

Evaluation



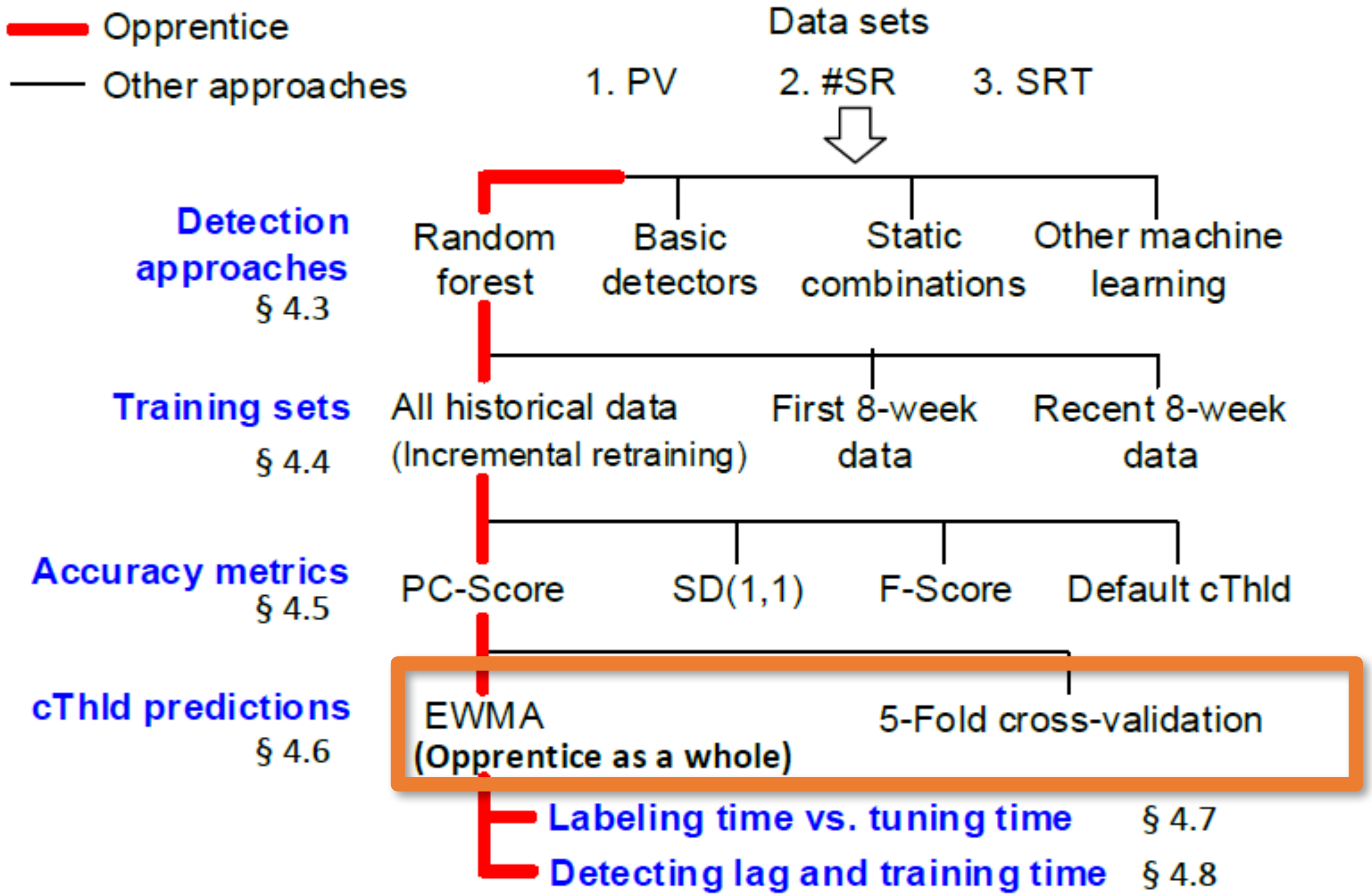
Opprentice
Other approaches
Data sets
1. PV
2. #SR
3. SRT
Detection approaches
§ 4.3
Random forest
Basic detectors
Static combinations
Other machine learning
Training sets
§ 4.4
All historical data (Incremental retraining)
First 8-week data
Recent 8-week data
Accuracy metrics
§ 4.5
PC-Score
SD(1,1)
F-Score
Default cThld
cThld predictions
§ 4.6
EWMA (Opprentice as a whole)
5-Fold cross-validation
Labeling time vs. tuning time
§ 4.7
Detecting lag and training time
§ 4.8

Random Forests vs. Other Learning Algorithms



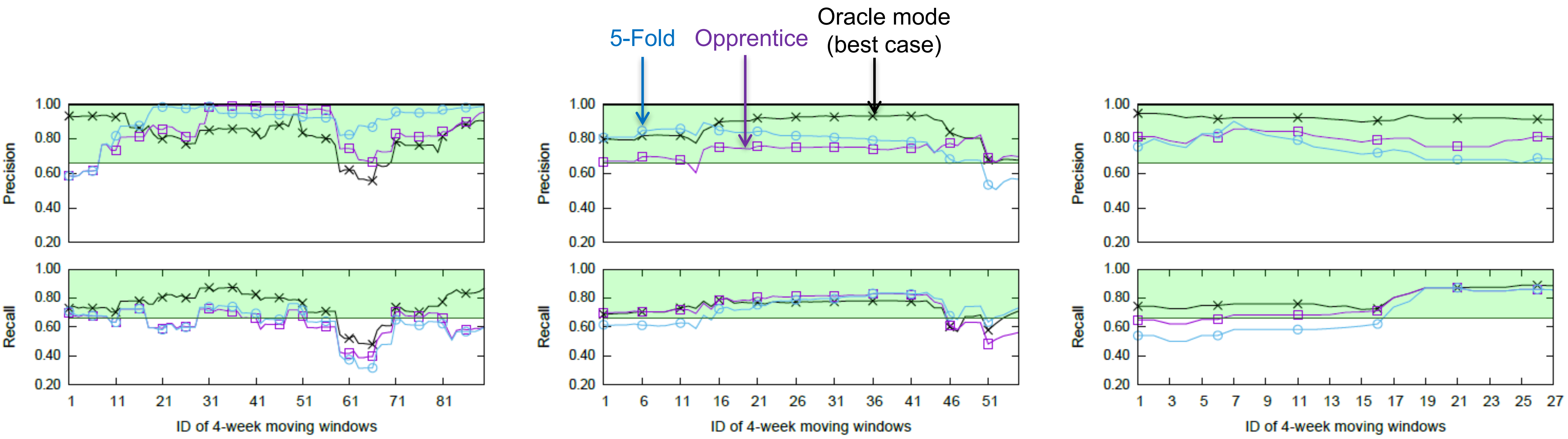
(The order of features is based on *mutual information*)

Evaluation



See the paper for full details

Opprentice as a whole



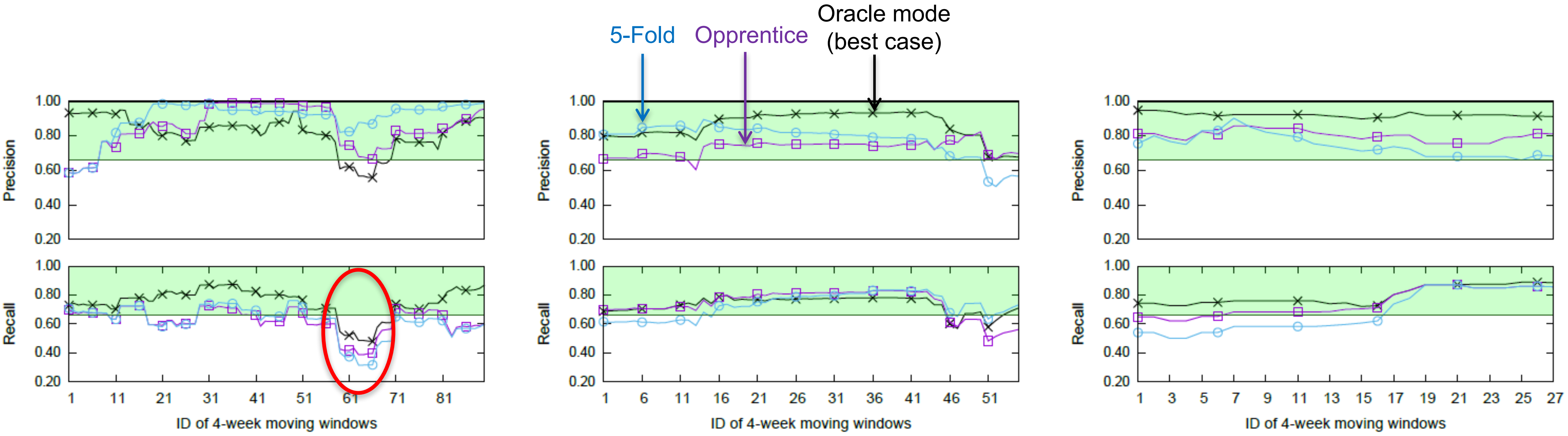
(a) KPI: PV (b) KPI: #SR (c) KPI: SRT

Opprentice achieves

40% **23%** **110%**

more points inside the preference regions than 5-Fold cross-validation

Opprentice as a whole



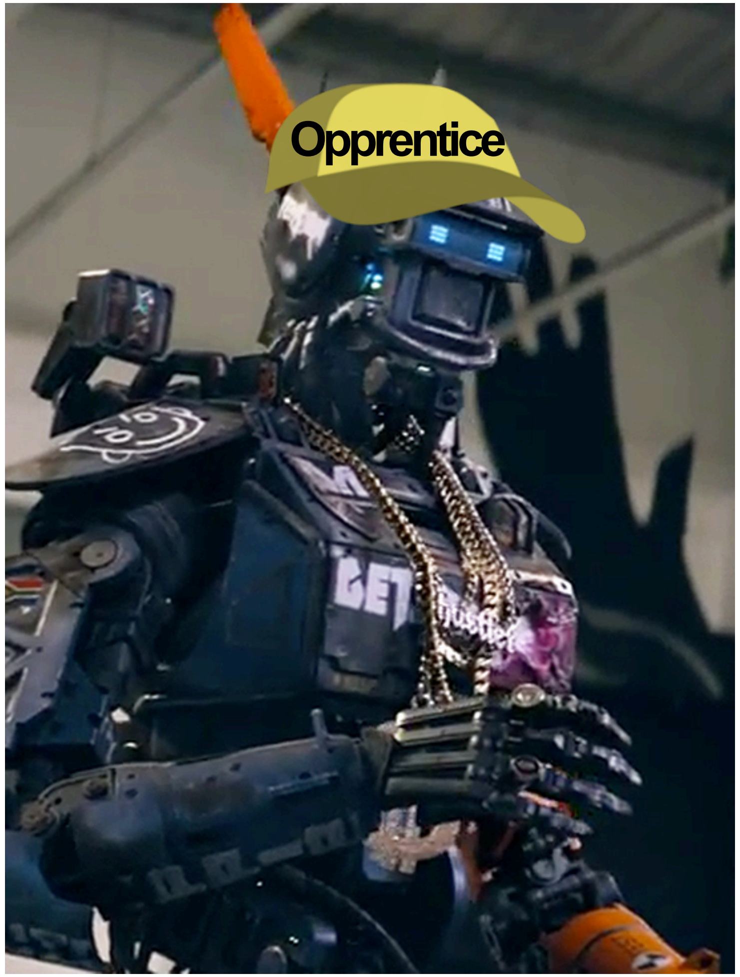
(a) KPI: PV (b) KPI: #SR (c) KPI: SRT

Opprentice achieves

40% **23%** **110%**

more points inside the preference regions than 5-Fold cross-validation

Conclusion



- Opprentice is an **automatic** and **accurate** machine learning framework for KPI anomaly detection

~~Defining anomalies~~ ~~Selecting detectors~~ ~~Tuning detectors~~

- Opprentice **bridges the gap** in applying complex detectors in practice

- The idea of Opprentice

  i.e., **using machine learning to model the domain knowledge**

could be a very promising way to automate other service managements

Thank you

liudp10@mails.tsinghua.edu.cn