CDMC 2019 Competition Winner Presentation

Masataka Kawai

Nara Institute of Science and Technology NAIST®

iplab Laboratory for Cyber Resilience

Data Mining Tasks

Task 1:

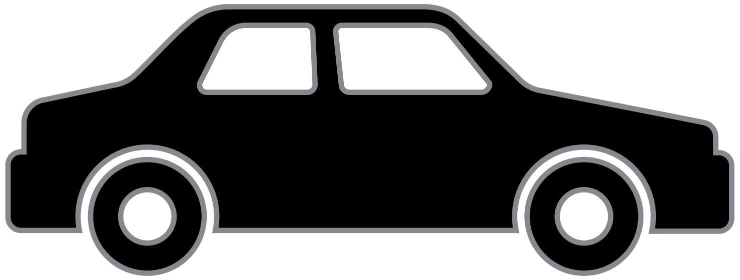
SADAVS-Sensor Array Data for Autonomous Vehicle Safety

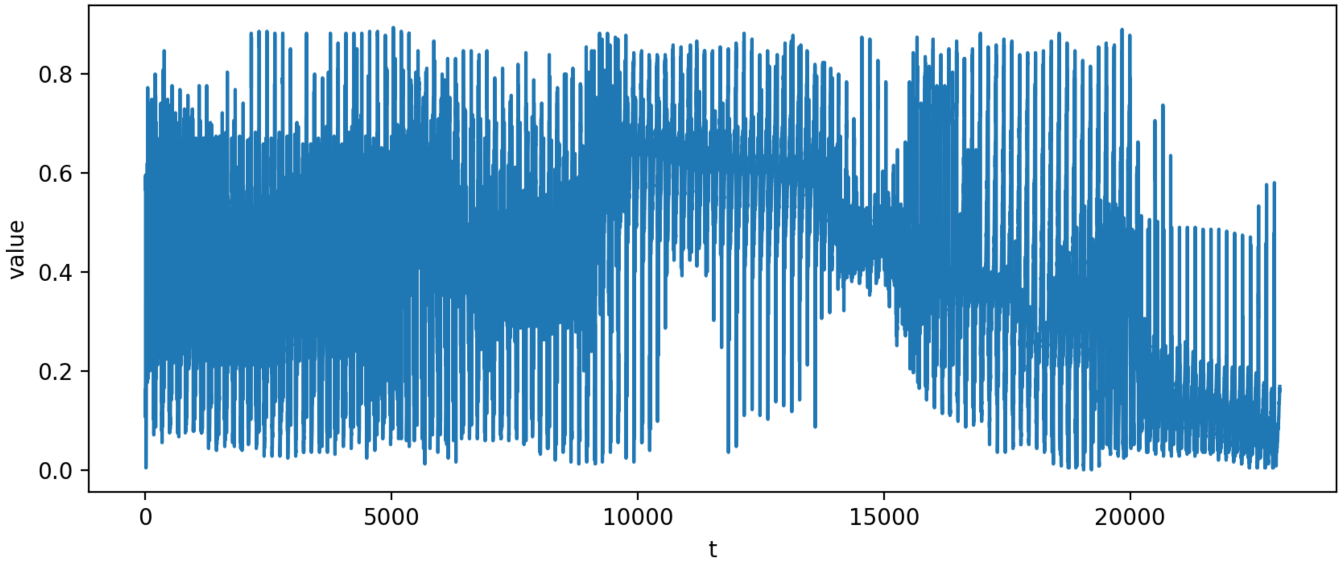
Task 2:

IoT malware classification

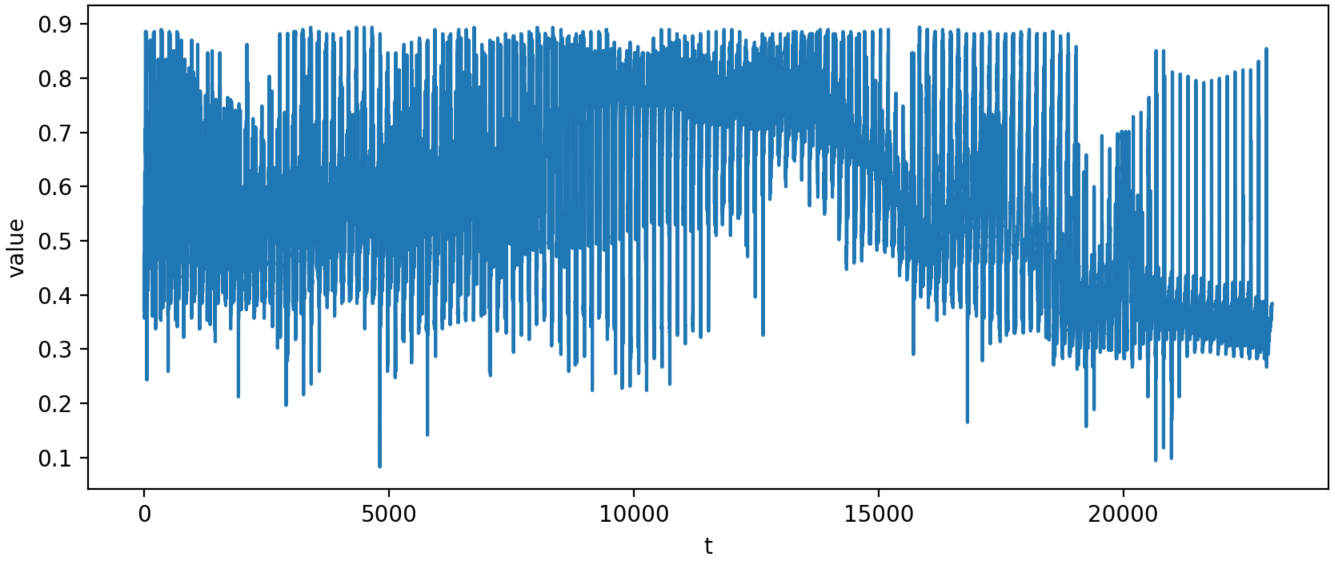
Task 1: SADAVS-Sensor Array Data for Autonomous Vehicle Safety

Purpose: Classify vehicle sensor data





Safe sensor data

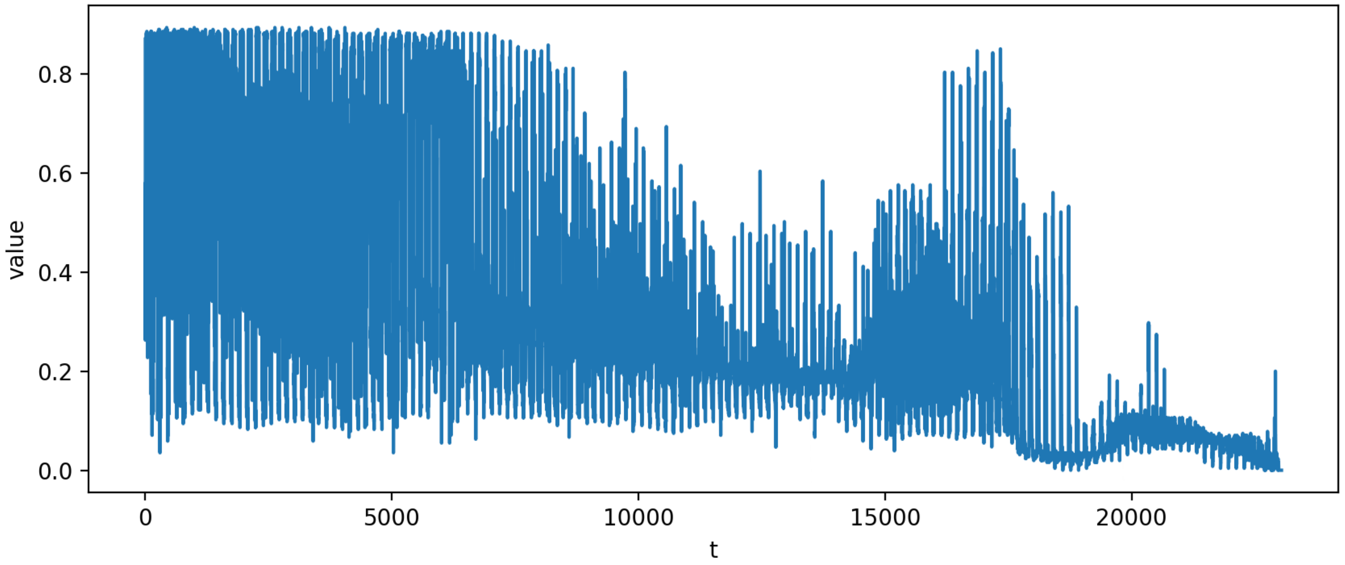


No-safe sensor data

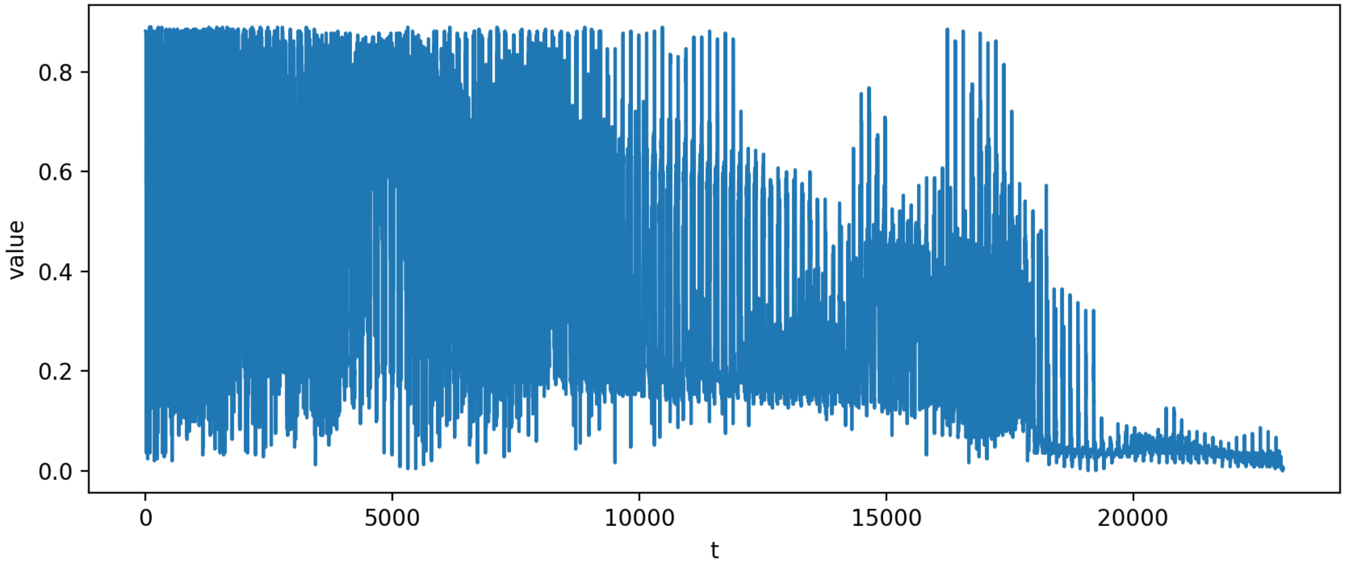
Task 1: SADAVS-Sensor Array Data for Autonomous Vehicle Safety

Test data has two scenarios

Scenario A



Scenario B



Task 1: SADAVS-Sensor Array Data for Autonomous Vehicle Safety

Candidate classification algorithms:

RF and XGBoost (choose better one)

Feature extraction method:

- Using the sensor data values as sequence data

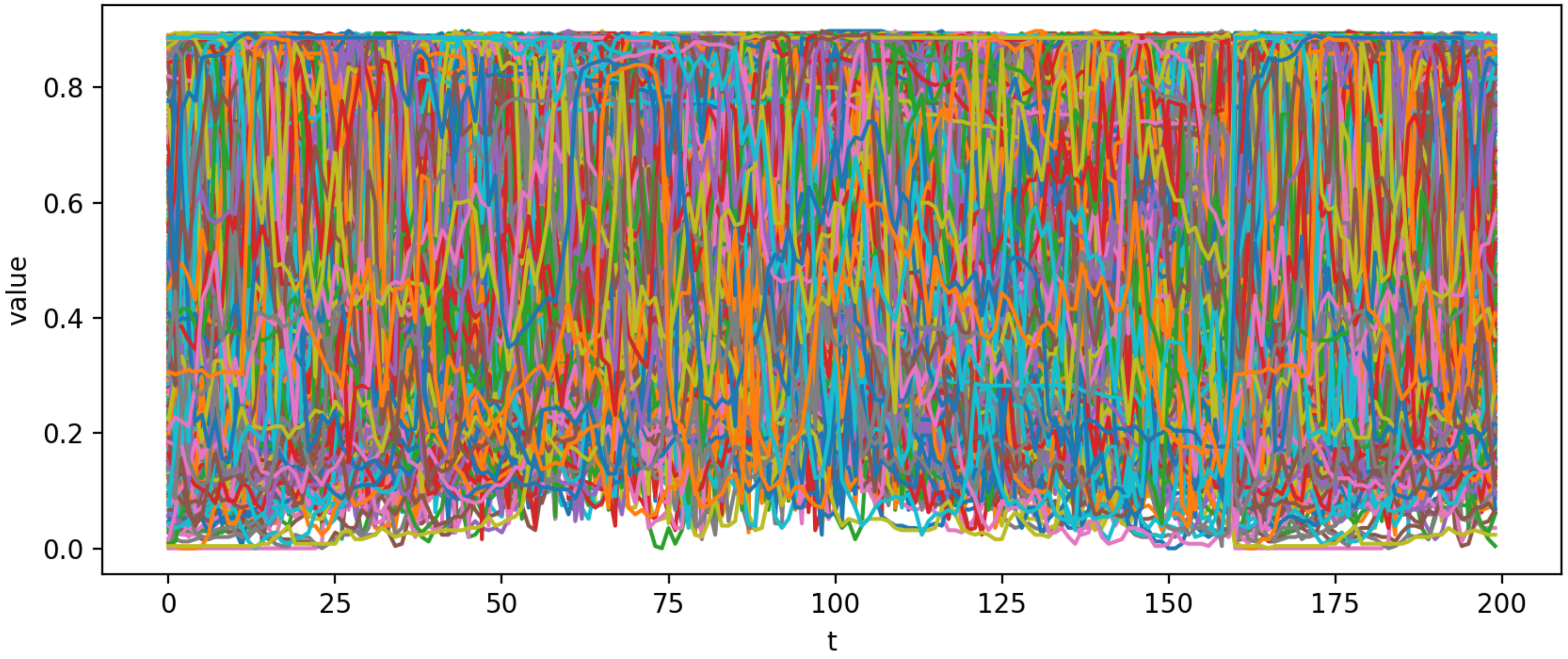
time data	1	2	3	4	…
1	0.1608	0.1059	0.3608	0.575	…

- Counting the frequency of each sensor data values

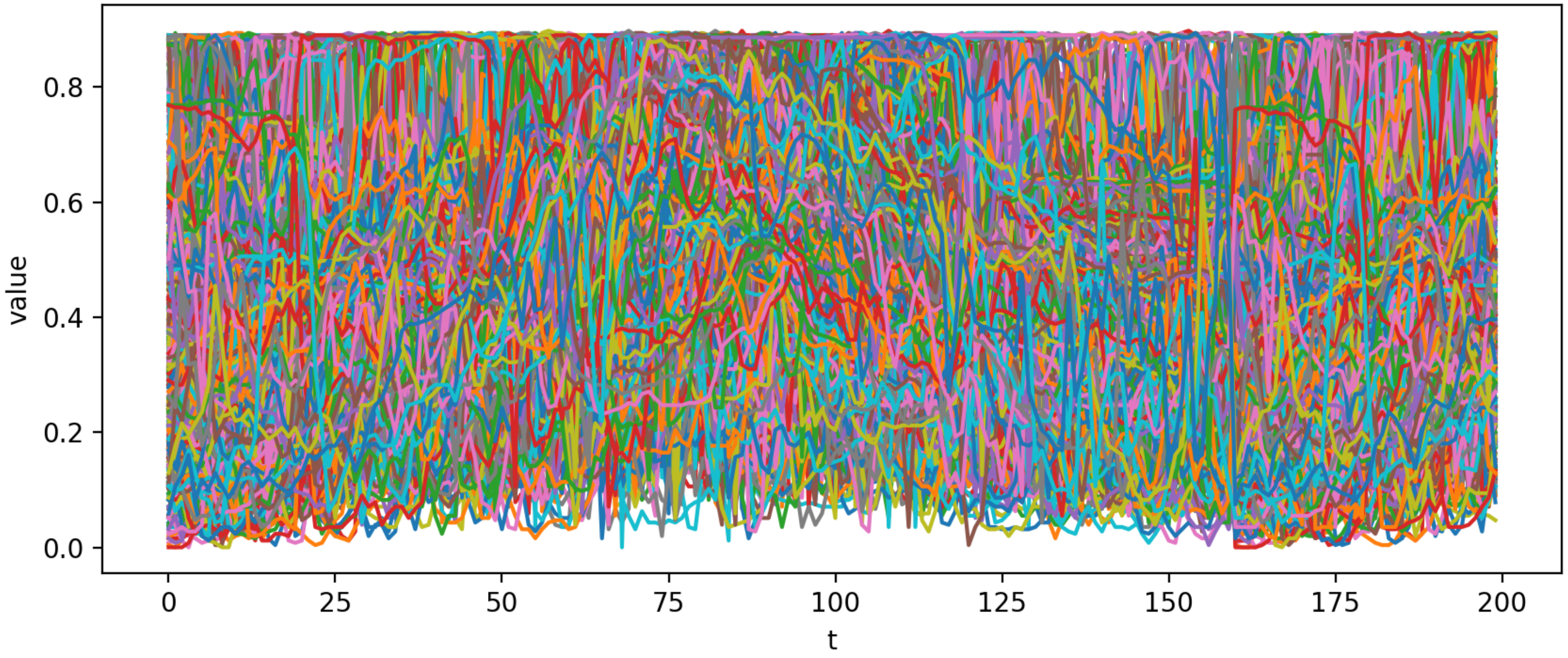
value data	0.1608	0.1059	0.3608	0.575	…
1	107	284	112	136	…

Task 1: SADAVS-Sensor Array Data for Autonomous Vehicle Safety

All safe training data



All no-safe training data



Task 1: SADAVS-Sensor Array Data for Autonomous Vehicle Safety

Evaluation:

Train (70% of the training data)

Test   (30% of the training data)

Accuracy (%):

ML algorithm Extraction method	RF	XGBoost
Counting the frequency	74.7	73.6

Task 1: SADAVS-Sensor Array Data for Autonomous Vehicle Safety

Summary

- Using the counting frequency feature extraction method
- Trained the machine using Random Forest
- Used all the training data

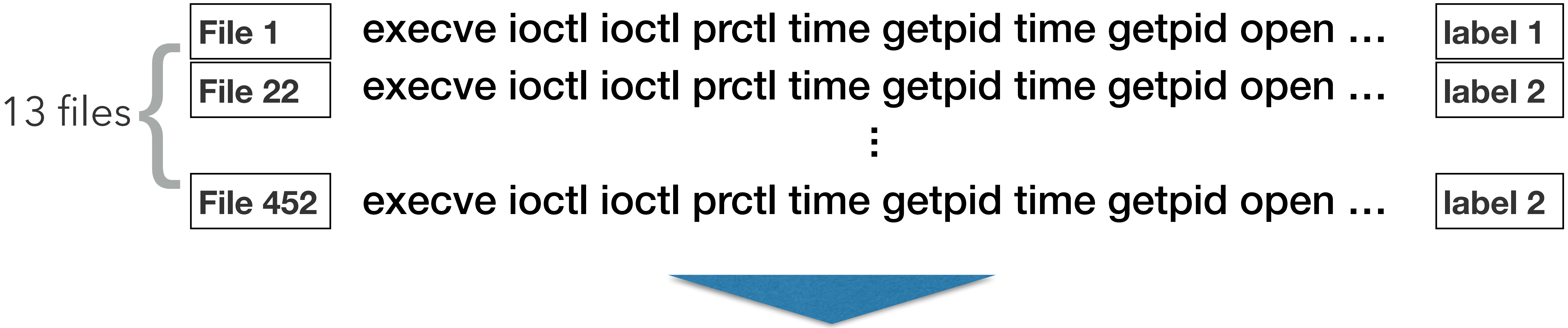
Task2: IoT malware classification

Purpose: Classify IoT malware

Execute in the sandbox and trace system call

```
execve ioctl ioctl prctl gettimeofday
getpid gettimeofday getpid fork
wait4 SIGCHLD exit EXIT
fork exit EXIT
chdir setuid32 setresuid32
rt_sigaction fork exit EXIT
socket fcntl fcntl connect _newselect
getsockopt rt_sigaction nanosleep
fork exit EXIT
close socket fcntl fcntl connect
_newselect getsockopt rt_sigaction
nanosleep fork exit EXIT
close socket fcntl fcntl connect
_newselect getsockopt rt_sigaction
nanosleep fork exit EXIT
⋮
```

Task2: IoT malware classification



13 files

File 1 execve ioctl ioctl prctl time getpid time getpid open … label 1

File 22 execve ioctl ioctl prctl time getpid time getpid open … label 2

⋮

File 452 execve ioctl ioctl prctl time getpid time getpid open … label 2

Task2: IoT malware classification

13 files

File 1	execve ioctl ioctl prctl time getpid time getpid open …	label 1
File 22	execve ioctl ioctl prctl time getpid time getpid open …	label 2
	⋮	
File 452	execve ioctl ioctl prctl time getpid time getpid open …	label 2

data \ label	1	2	3	4	5
execve ioctl ioctl prctl time getpid time getpid open …	2	11	0	0	0

Correct label is 2.

Combine 13 files into 1 file with the correct label.

Task2: IoT malware classification

Candidate Classification algorithms:

RF and XGBoost (choose better one)

Feature extraction methods:

- Counting the frequency of each system call

Accuracy (%):

ML algorithm Extraction method	RF	XGBoost
Counting the frequency of each system call	98.1	97.6

Task2: IoT malware classification

Summary

- Counted and shorten the system call data
- Using the count method for feature extraction
- Trained the machine using Random Forest
- Used all the training data