Machine learning for computer security

Junichi Murakami
Executive Officer, Director of Advanced Development Division

FFRI,Inc.
http://www.ffri.jp

Agenda

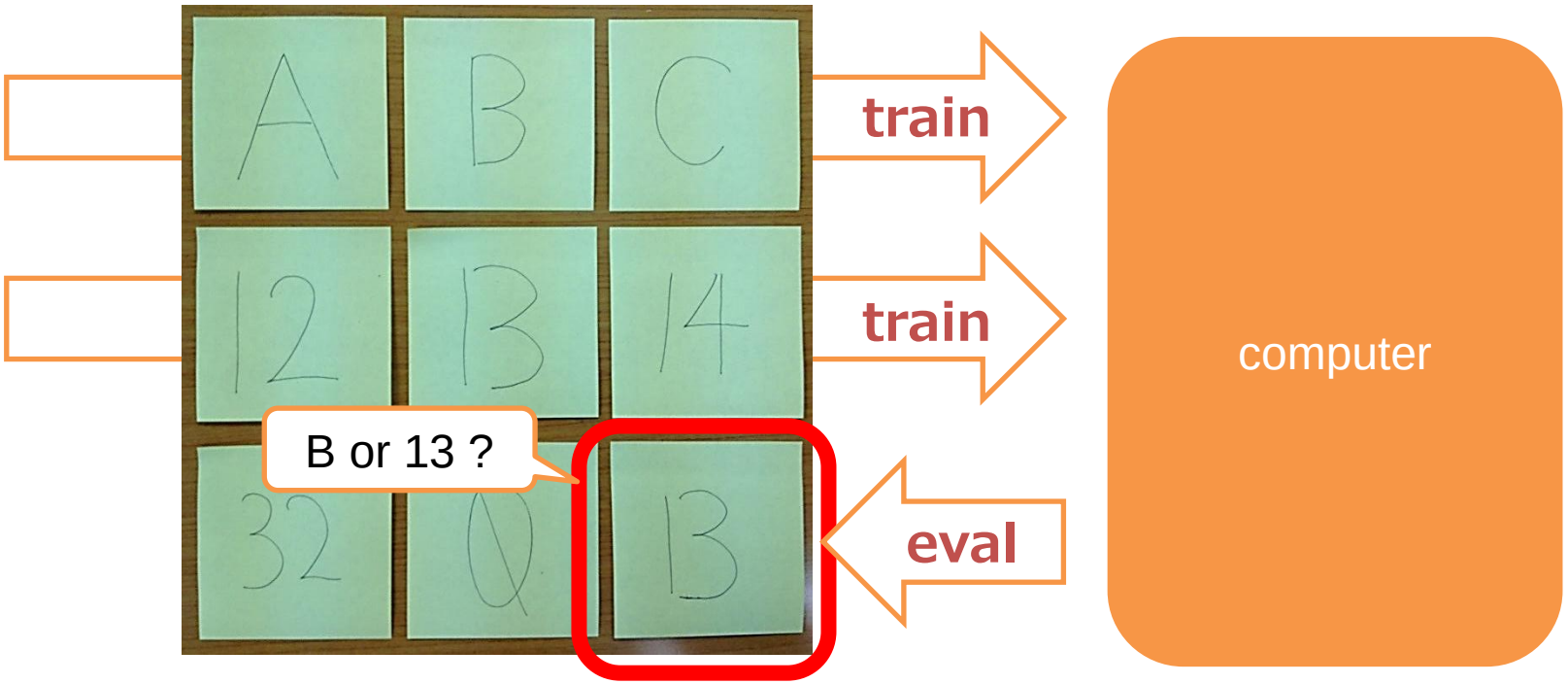
1. Introduction
2. Machine learning basics
   - What is “Machine learning”
   - Background and circumstances
   - Type of machine learning
   - Implementations
3. Malware detection based on machine learning
   - Overview
   - Datasets
   - Cuckoo Sandbox
   - Jubatus
   - Evaluations
   - Possibility for another applications
4. Conclusions
5. References

Introduction

- First this slides describes a basis of machine learning then introduces malware detection based on machine learning
- The Author is a security researcher, but is not an expert of machine learning
- Currently the malware detection is experimental

What is “Machine learning”

- By training a computer, letting the computer estimate and predict something based on the experience
- Based on artificial intelligence research



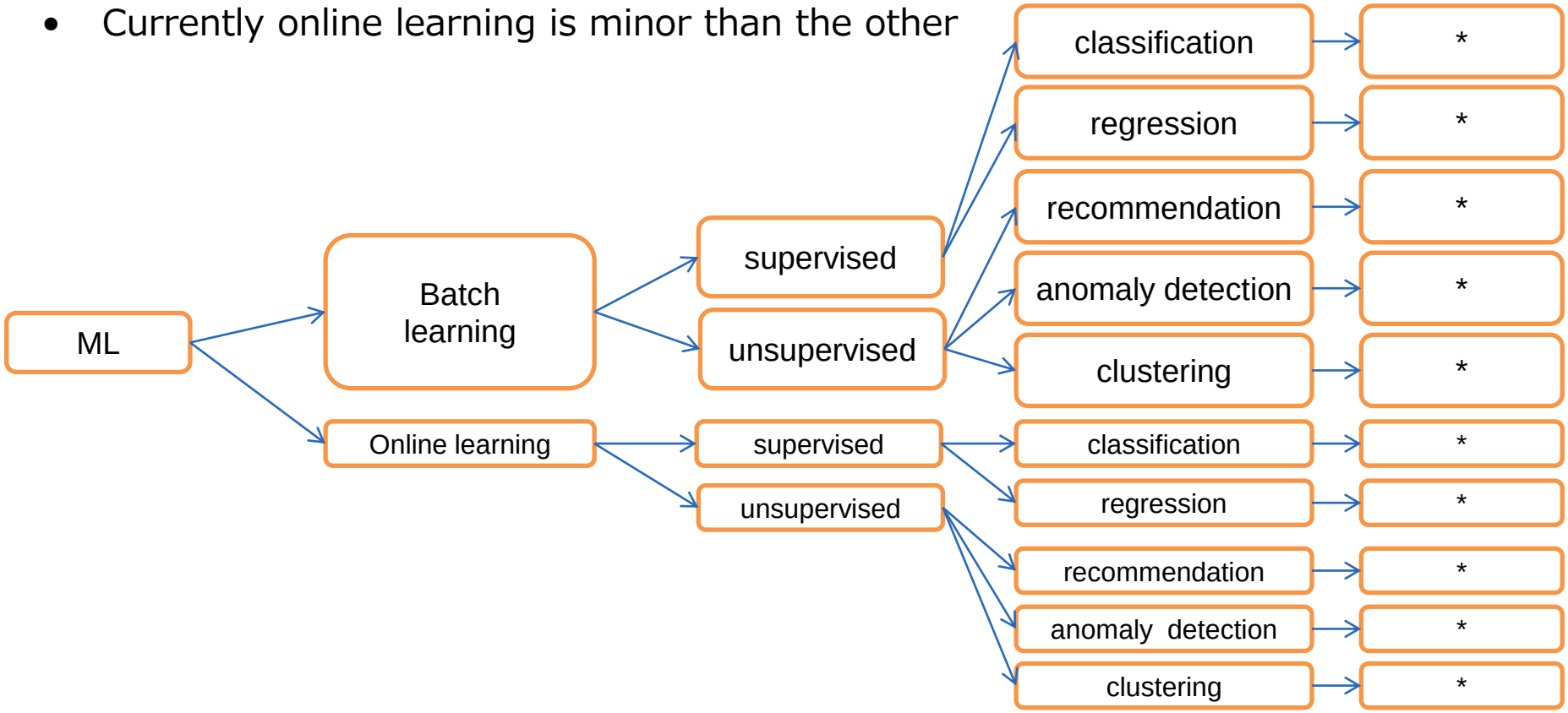
Background and circumstances

- Recently, big data analysis attracts attention
  - E-Commerce, Online games, BI（Business Intelligence), etc.
  - The demand would probably increase more through spreading the M2M

- Machine learning as method for big data analysis
  - Analyze data and estimate the future
  - Penetration rate is different by each industry
  - In IT security, not enough used yet

Type of machine learning

- “Machine learning” is general word which contains various themes and methods
- Roughly it can be classified as shown below
- Currently online learning is minor than the other



Type of machine learning(cont.)

- Batch learning
  - processing all stocked data at once
- Online learning
  - processing data one after another (new data comes successively)

- Supervised: train by labeled data
  - classification, regression, recommendation, etc.
    eg. apple=fruit, tomato=vegetable, pineapple=???
- Unsupervised : train by non-labeled data
  - clustering(grouping by similarity),
    anomaly detection(detect outlier, changing point, etc.)

Implementations

- The specific method of each tasks are mainly researched on academic(#)
  (This slides doesn't mention them)

- Examples of available frameworks and libraries
  - Hadoop
    - Hadoop doesn't have machine learning function but Hadoop-based 3rd party frameworks are available
  - Apache Mahout
    - Representative of Hadoop-based framework above
    - Supports clustering, classification, etc. with batch learning
  - Jubatus
    - Distributed online machine learning framework(mention later)
  - Other libraries
    - Libsvm - http://www.csie.ntu.edu.tw/~cjlin/libsvm/
    - dlib ml - http://dlib.net/ml.html
    - Shark - http://shark-project.sourceforge.net/

eg. ICML- http://icml.cc/2013/

Malware detection based on machine learning

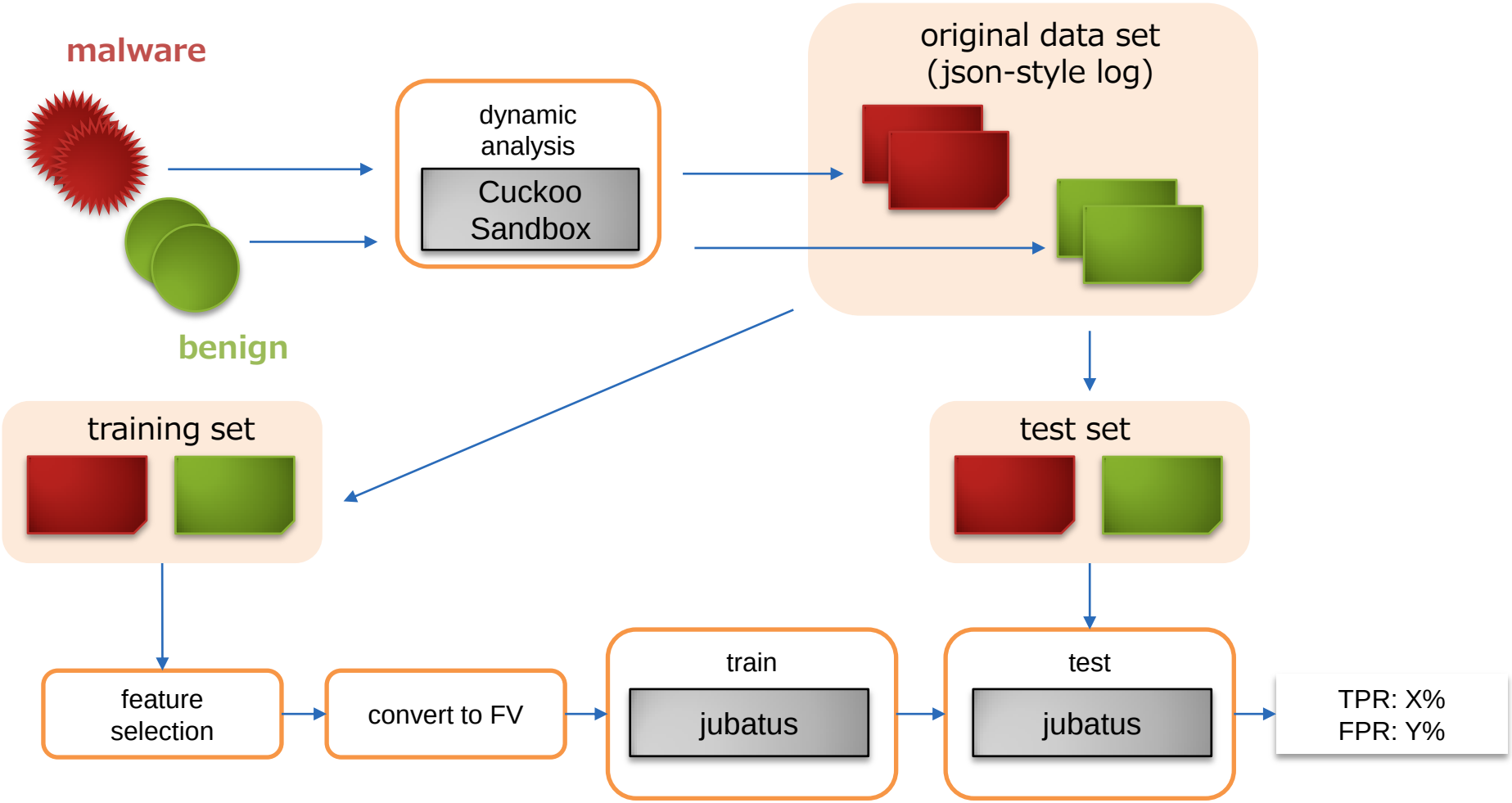
- Making a malware detection based on classification
  - predict(check) if input program file is malware

- Steps
  1. Prepare training set and test set for malware and benign respectively

  2. Select features from datasets, and convert them to feature vectors(FV)
     - Basically, the task of expert about applied field
     - Select features based on each own experience and knowledge
     - We mainly extracted API calls log recorded by Cuckoo Sandbox

  3. Train by FV added label(malware or benign)
  4. Classify the test set by extracting FV from them
     - This time, we used Jubatus for machine learning framework

Overview



TPR: True positive ratio
FPR: False positive ratio

Datasets

- Malware / 2641 (training set : 1320 / test set : 1321)
  - Random sample from collected malware in latest 6 months
  - TPR is not good enough based on metascan(#) results (over 10AVVs)
    - Average TPR: about 30%, best TPR: about 60%

- Benign / 1803 (training set : 893 / test set : 910)

http://www.opswat.com/products/metascan

Cuckoo Sandbox - http://www.cuckoosandbox.org

- Open source automated malware analysis system
  - Execute malware inside virtual machine
  - Monitoring its behavior in runtime
  - Collaborate with VirusTotal(Hash search) and yara
- Execute each sample within 90 seconds
- Extract API calls and other information from log, then convert to FV

Cuckoo Sandbox (ex. API log)


```
"calls": [
    {
        "category": "system",
        "status": "FAILURE",
        "return": "0xc0000135",
        "timestamp": "2013-02-28 12:03:49,478",
        "thread_id": "420",
        "repeated": 0,
        "api": "LdrLoadDll",
        "arguments": [
            { "name": "Flags", "value": "1242916" },
            { "name": "FileName", "value": "C:¥¥WINDOWS¥¥system32¥¥VB6JP.DLL" },
            { "name": "BaseAddress", "value": "0x00000000" }
        ]
    },
    {
        "category": "registry",
        "status": "SUCCESS",
        "return": "0x00000000",
        "timestamp": "2013-02-28 12:03:49,528",
        "thread_id": "420",
        "repeated": 0,
        "api": "NtOpenKey",
        "arguments": [
            { "name": "KeyHandle", "value": "0x00000058" },
            { "name": "DesiredAccess", "value": "1" },
            { "name": "ObjectAttributes", "value": "Registry¥¥MACHINE¥¥System¥¥Curre
        ]
    },
```

Jubatus – http://jubat.us/en

- Developed by Preferred Infrastructure, Inc. and NTT Software Innovation Center
- Latest version is 0.4.4 (21/06/2013)
  - 1st release: 0.1.0(26/10/2011)
- Open source, LGPL v2.1
- Distributed online machine learning framework
  - Can analyze daily collected malware and monitor those trend continuously
  - Can be scaled out by adding server
- Supports various machine learning
  - Classification, Regression, Recommendation, Anomaly Detection, etc.
- C++, Python, Ruby and Java bindings are available

Evaluations

- TPR and FPR are fluctuated depending type of FV and various parameters
- Current (Jun 2013) best result is shown as below
  - By comparison with traditional method like pattern matching, high TPR is confirmed
  - theoretically, FP would occur 8 files out of 1000 files (needs to be improved)
- Datasets is limited, so additional test is required

malware(files)	benign(files)	TPR(%)	FPR(%)
1321	910	94.5495	0.8791

(Jun 2013)

Possibility of another applications

- Possible to apply to various field if we can collect data
  - Classification x malware -> detection(this trial)
  - Clustering x malware -> family analysis and new family detection
  - Anomaly detection x network traffic -> C&C detection
  - Anomaly detection x traffic pattern -> infected host detection
  - Anomaly detection x authentication log -> spoofing detection
  - and more...

Conclusions

- Machine learning is attracted a great deal of attention as a method of big data analysis
- Can apply to security technology
  - malware detection and other possibility are available
- Tried to make a malware detection experimentally
  - Leaving some problems, but we can expect it to be applied to detecting recent advanced malware

References

- Jubatus
  - http://jubat.us/en/
- Machine learning tutorials
  - http://www.slideshare.net/unnonouno/jubatus-casual-talks
- Automatic Analysis of Malware Behavior using Machine Learning
  - http://pi1.informatik.uni-mannheim.de/malheur/