ANM 2019 Fall
Assignments and Project

Nengwen Zhao

znw17@mails.tsinghua.edu.cn

Overview

- Assignment #1: Data Preprocessing and Visualization (10%)
- Assignment #2: Log Analysis for Anomaly Detection (20%)
- Project: Time Series Anomaly Detection Algorithm Competition (60%)

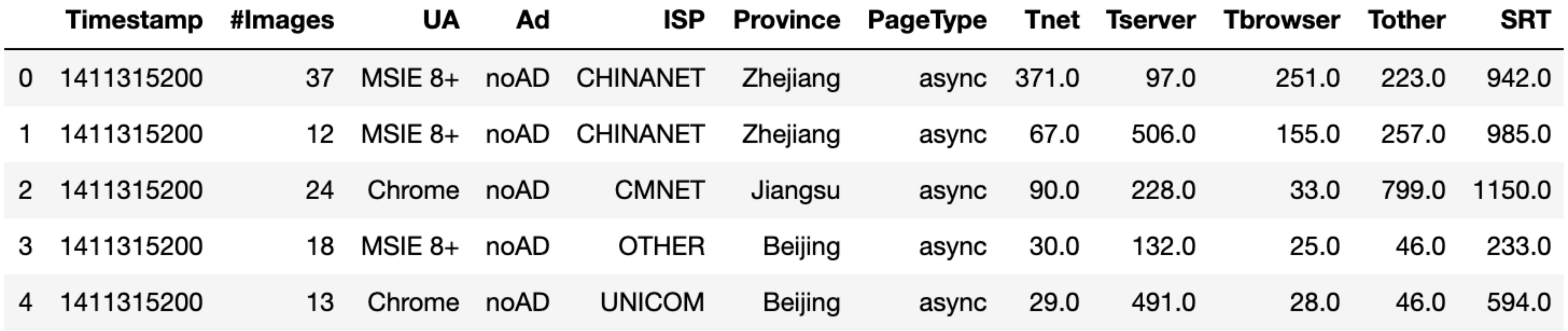
Each student finishes the assignment alone and a team of 2-3 students finish the project together.

What you can learn

- At least one programming language (Python is recommended).
- At least one data visualization tool, such as PowerBI, Tableau, Matlab, and Matplotlib.
- Some background regarding AIOps (Artificial Intelligence for IT Operations).
- Some machine learning tools such as Scikit-learn, PyTorch, TensorFlow, Keras.

Assignment #1-- Data Preprocessing and Visualization

- 2 weeks of search logs from a global top search engine
- Basic statistics: coding to count the data in different ways, e.g., how many queries are served per minute. You can also use Power BI do it.
- Visualization: plot figures to show the data (e.g., line chart, histogram). You can use Power BI, Tableau, Matplotlib, Matlab, etc. to do it.

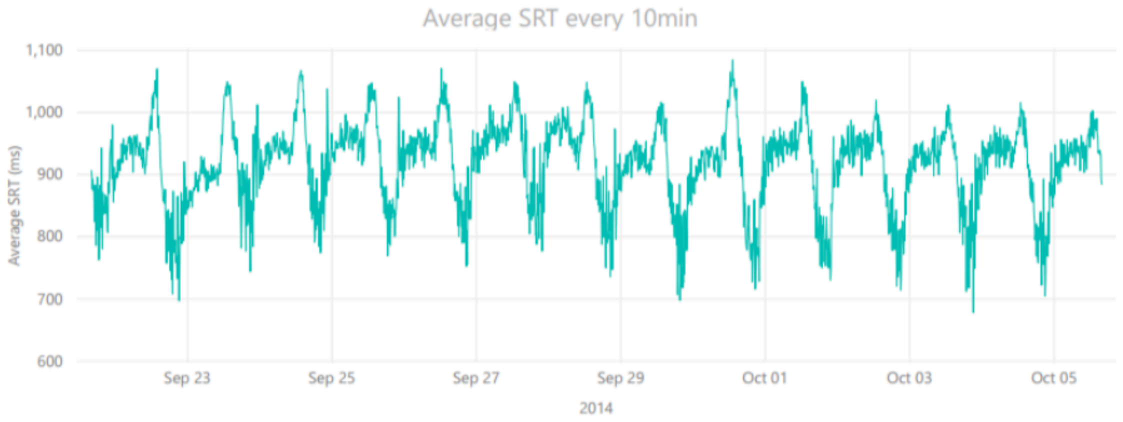


	Timestamp	#Images	UA	Ad	ISP	Province	PageType	Tnet	Tserver	Tbrowser	Tother	SRT
0	1411315200	37	MSIE 8+	noAD	CHINANET	Zhejiang	async	371.0	97.0	251.0	223.0	942.0
1	1411315200	12	MSIE 8+	noAD	CHINANET	Zhejiang	async	67.0	506.0	155.0	257.0	985.0
2	1411315200	24	Chrome	noAD	CMNET	Jiangsu	async	90.0	228.0	33.0	799.0	1150.0
3	1411315200	18	MSIE 8+	noAD	OTHER	Beijing	async	30.0	132.0	25.0	46.0	233.0
4	1411315200	13	Chrome	noAD	UNICOM	Beijing	async	29.0	491.0	28.0	46.0	594.0

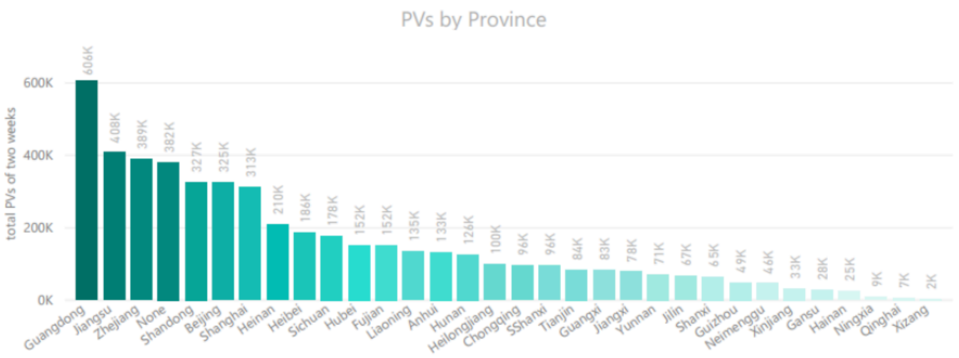
Assignment #1-- Data Preprocessing and Visualization

- Calculate the average SRT of every 10 minutes, and plot the SRT with a line chart (x axis for date time and y axis for the average SRT).
- Calculate the average of each SRT component of every 10 minute, and plot the four SRT components together with a stacked area chart (x axis for date time and y axis for time) and also a 100% stacked area chart (y axis for the percentage).
- Plot the CDF (Cumulative distribution function) chart of SRT.
- Plot the CDF chart of #Images.
- Count the number of queries (also called page views or PVs) of each minute, and plot the minute-level PVs with a line chart (x axis for date time and y axis for the PVs).
- Count the PVs of each province, and plot it with a histogram chart (x axis for province and y axis for PVs).
- Count the PVs of each UA, and plot it with a pie chart (show the percentages in the chart).
- What are the differences among those charts (How to decide which one to use)
- Describe your experience or findings in doing those jobs. For example, experience of processing the data, observations from the charts, characteristics of the data, potential explanations, and any interesting things you would like to mention.

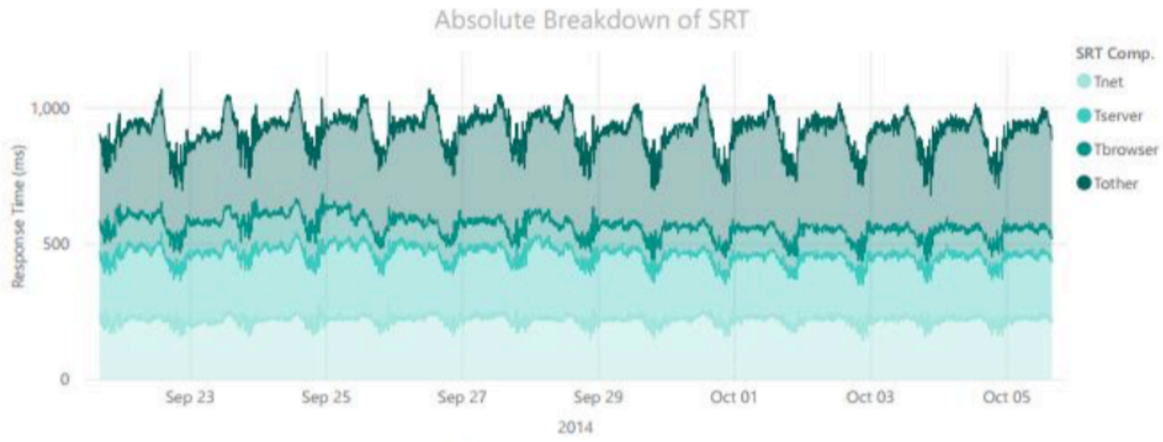
Assignment #1-- Data Preprocessing and Visualization



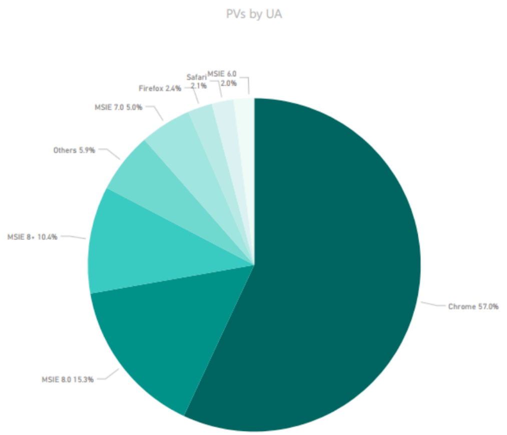
Average SRT every 10min



PVs by Province



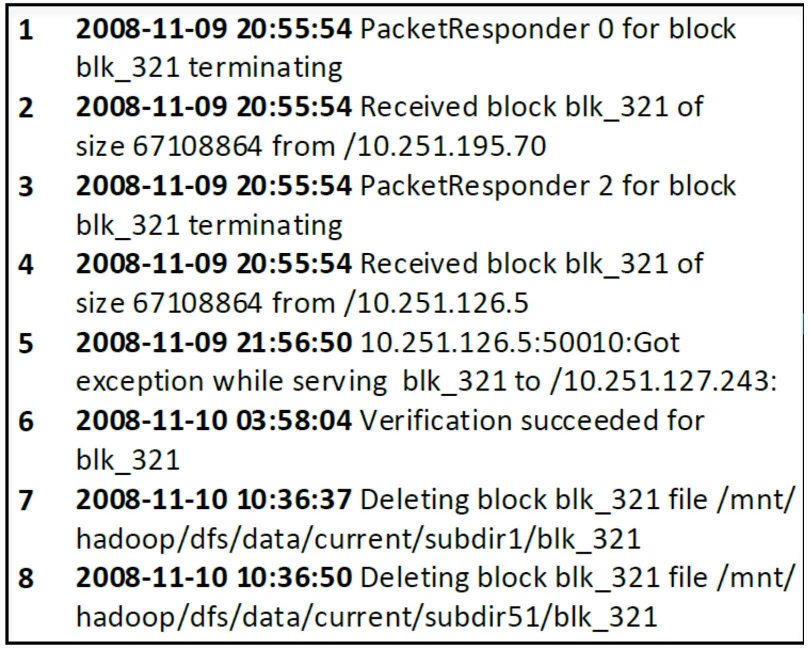
Absolute Breakdown of SRT



PVs by UA

Assignment #2– Log Analysis for Anomaly Detection

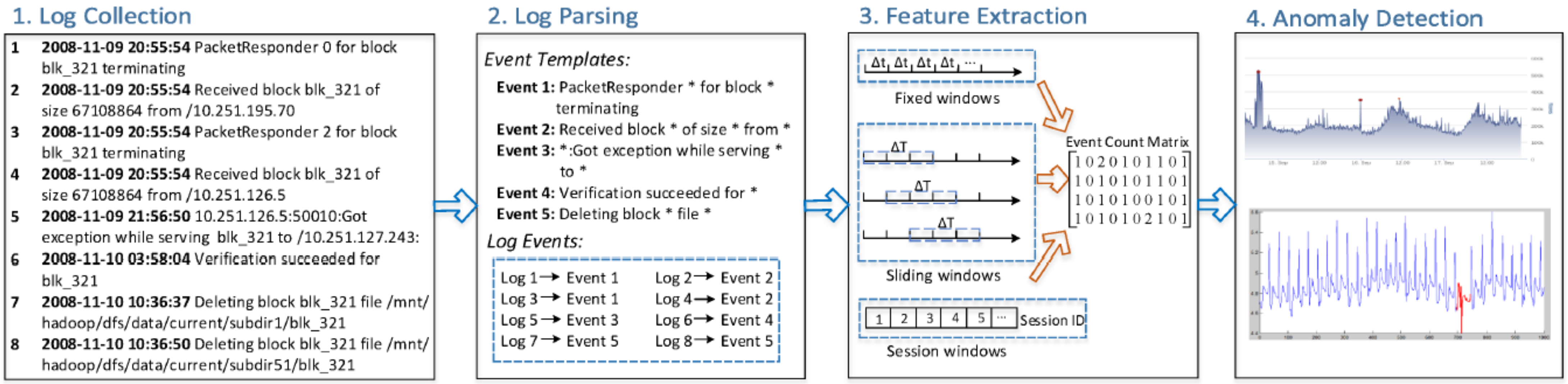
- Logs are the main data source for system anomaly detection.
- Logs are routinely generated by systems (e.g., 24 x 7 basis).
- Logs record detailed runtime information, e.g., timestamp, state, IP address.



```
1   2008-11-09 20:55:54 PacketResponder 0 for block
    blk_321 terminating
2   2008-11-09 20:55:54 Received block blk_321 of
    size 67108864 from /10.251.195.70
3   2008-11-09 20:55:54 PacketResponder 2 for block
    blk_321 terminating
4   2008-11-09 20:55:54 Received block blk_321 of
    size 67108864 from /10.251.126.5
5   2008-11-09 21:56:50 10.251.126.5:50010:Got
    exception while serving  blk_321 to /10.251.127.243:
6   2008-11-10 03:58:04 Verification succeeded for
    blk_321
7   2008-11-10 10:36:37 Deleting block blk_321 file /mnt/
    hadoop/dfs/data/current/subdir1/blk_321
8   2008-11-10 10:36:50 Deleting block blk_321 file /mnt/
    hadoop/dfs/data/current/subdir51/blk_321
```

Assignment #2– Log Analysis for Anomaly Detection

Popular Framework of log anomaly detection



```
1   2008-11-09 20:55:54 PacketResponder 0 for block
    blk_321 terminating
2   2008-11-09 20:55:54 Received block blk_321 of
    size 67108864 from /10.251.195.70
3   2008-11-09 20:55:54 PacketResponder 2 for block
    blk_321 terminating
4   2008-11-09 20:55:54 Received block blk_321 of
    size 67108864 from /10.251.126.5
5   2008-11-09 21:56:50 10.251.126.5:50010:Got
    exception while serving  blk_321 to /10.251.127.243:
6   2008-11-10 03:58:04 Verification succeeded for
    blk_321
7   2008-11-10 10:36:37 Deleting block blk_321 file /mnt/
    hadoop/dfs/data/current/subdir1/blk_321
8   2008-11-10 10:36:50 Deleting block blk_321 file /mnt/
    hadoop/dfs/data/current/subdir51/blk_321
```

Event Templates:

Event 1: PacketResponder * for block * terminating
Event 2: Received block * of size * from *
Event 3: *:Got exception while serving * to *
Event 4: Verification succeeded for *
Event 5: Deleting block * file *

Log Events:

Log 1→ Event 1 Log 2→ Event 2
Log 3→ Event 1 Log 4→ Event 2
Log 5→ Event 3 Log 6→ Event 4
Log 7→ Event 5 Log 8→ Event 5

$$\begin{bmatrix} 1&0&2&0&1&0&1&1&0&1 \\ 1&0&1&0&1&0&1&1&0&1 \\ 1&0&1&0&1&0&0&1&0&1 \\ 1&0&1&0&1&0&2&1&0&1 \end{bmatrix}$$

Current log parsing methods:

- **LogSig (CIKM’11)**
- **LKE (ICDM’09)**
- **IPLoM (KDD’09)**
- **SLCT (IPOM’03)**

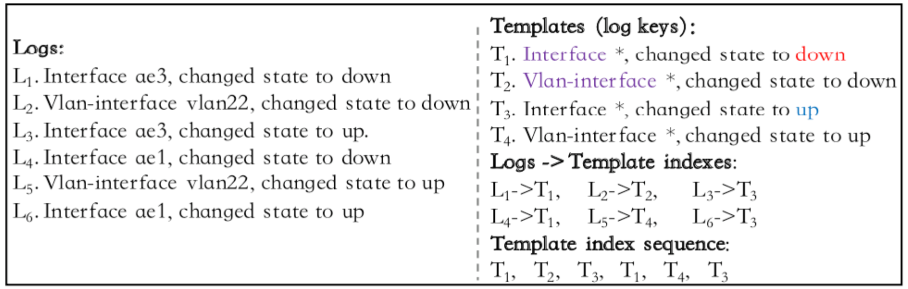
Current anomaly detection methods:

- **Log Clustering (ICSE’17)**
- **PCA (SOSP’09)**
- **Invariants Mining (ATC’10)**

Assignment #2– Log Analysis for Anomaly Detection

Part 1: compare current log parsing methods

- Code:
  - https://github.com/logpai/logparser
  - Four algorithms: LogSig, IPLoM, SLCT, LKE



Logs:

L_1. Interface ae3, changed state to down

L_2. Vlan-interface vlan22, changed state to down

L_3. Interface ae3, changed state to up.

L_4. Interface ae1, changed state to down

L_5. Vlan-interface vlan22, changed state to up

L_6. Interface ae1, changed state to up

Templates (log keys):

T_1. Interface *, changed state to down

T_2. Vlan-interface *, changed state to down

T_3. Interface *, changed state to up

T_4. Vlan-interface *, changed state to up

Logs -> Template indexes:

L_1->T_1, L_2->T_2, L_3->T_3

L_4->T_1, L_5->T_4, L_6->T_3

Template index sequence:

T_1, T_2, T_3, T_1, T_4, T_3

- Data:
  - https://github.com/logpai/logparser/tree/master/data
  - Five datasets (BGL, HDFS, HPC, Proxifier, Zookeeper).
- Requirement:
  1. Run four algorithms (LogSig, IPLoM, SLCT, LKE).
  2. Compare the running time, F-score, RandIndex respectively when four algorithms parse five datasets.

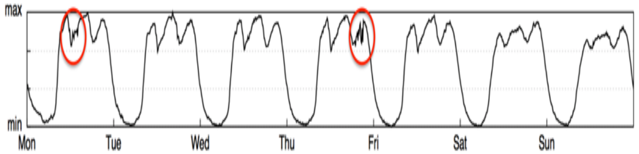
Assignment #2– Log Analysis for Anomaly Detection

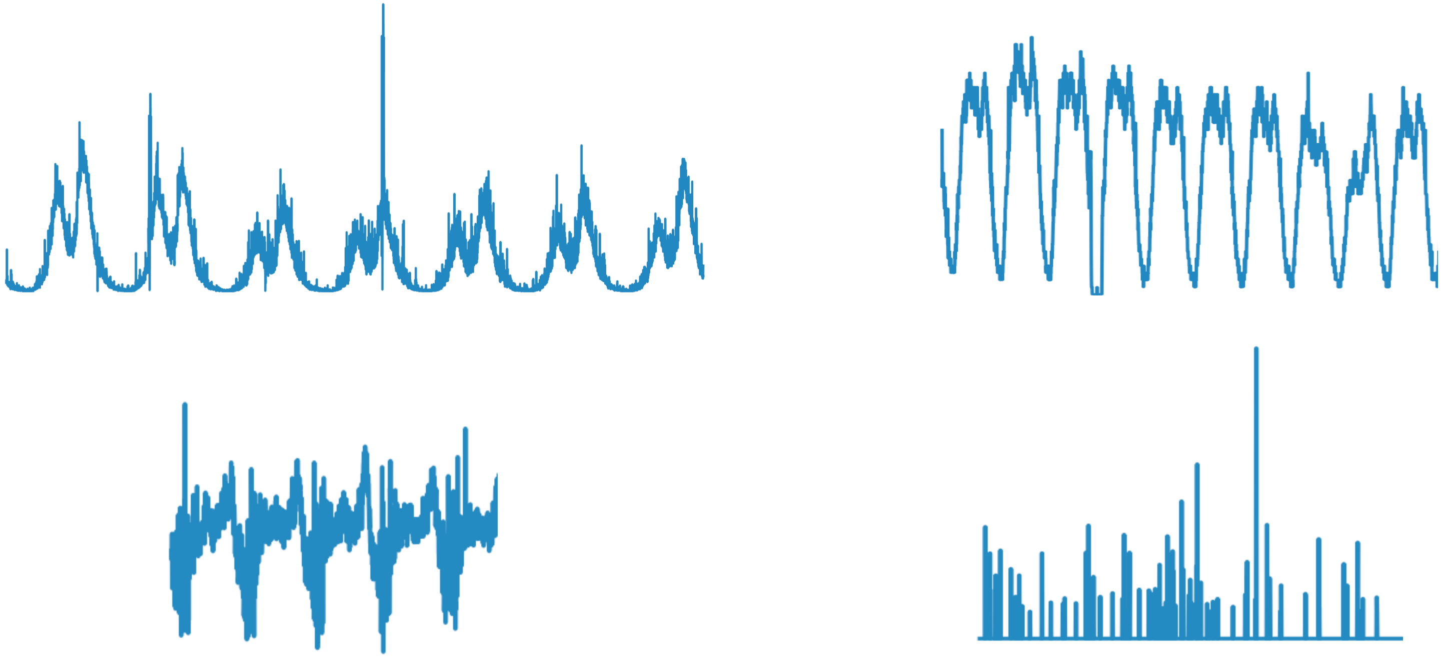
Part 2: compare anomaly detection methods

- Code:
  - https://github.com/logpai/loglizer
  - Algorithms: Invariants Mining, PCA and Log Clustering
- Data:
  - HDFS logs with labels (1.5G)
- Requirement:
  - Choose a log parsing algorithm to change HDFS logs into template sequence, then run three different anomaly detection methods.
  - Display precision, recall, F-score, and running time
  - Run *invariants mining* and display three invariants.

Project – Time Series Anomaly Detection

- Anomaly detection: binary classification problem
- Dataset: 26 labeled KPIs from five large Internet companies





Some examples of the dataset

Project – Time Series Anomaly Detection

- Training set: 50%, with label used to train your algorithm
- Testing set: 50%, without label used to test your algorithm

KPI ID	timestamp	value	label
A	1411315200	90.75	0
A	1411315260	96.78	1
…	…	…	

Training set

KPI ID	timestamp	value
A	1411423000	83.2
A	1411423060	91.4
…	…	…

Testing set

KPI ID	timestamp	predict
A	1411423000	0
A	1411423060	1
…	…	

Submitted files

Project – Time Series Anomaly Detection

Requirements:

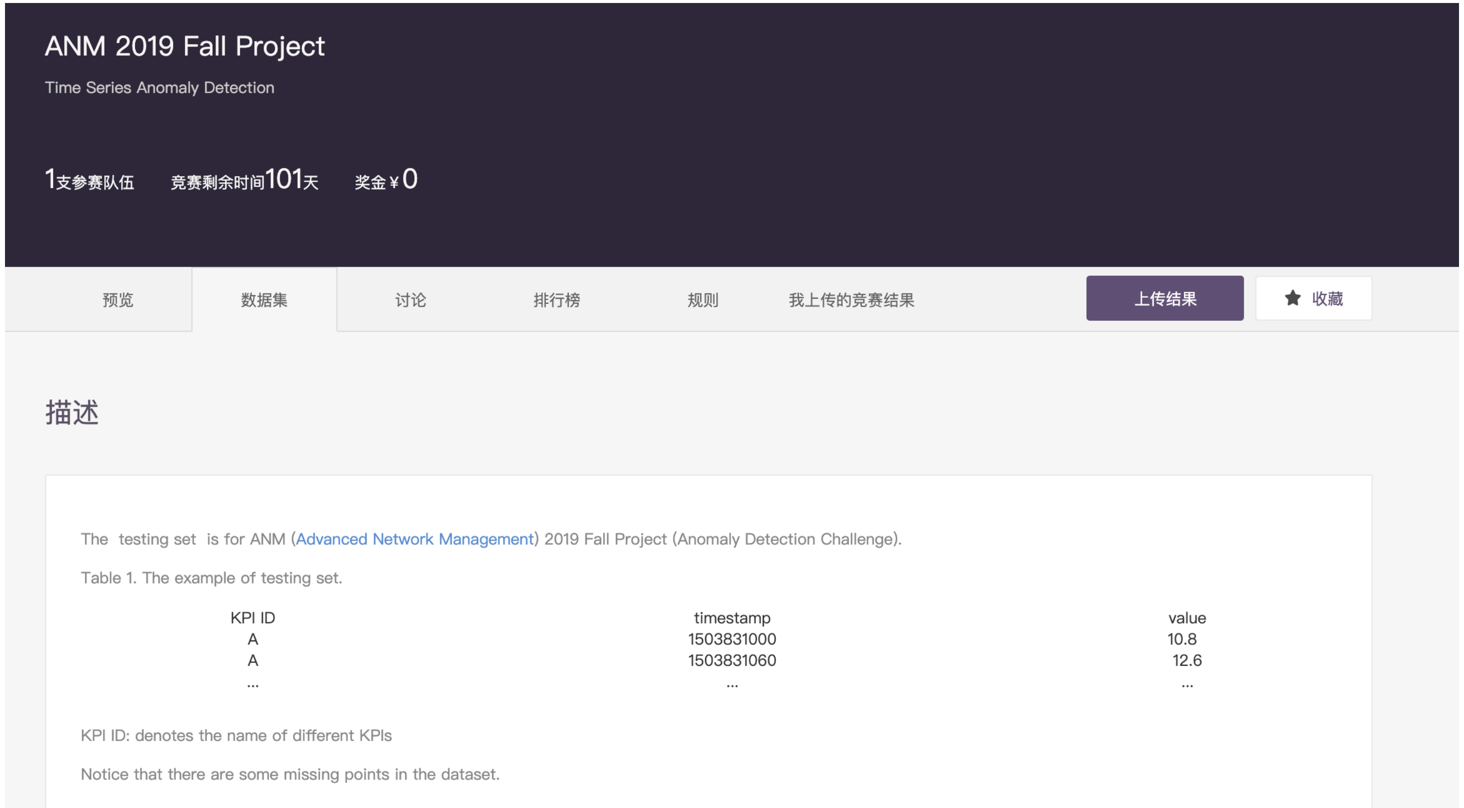
- Design a generic anomaly detection algorithm and submit your result on the website. The website will give a rank list of F-score like Kaggle.
- Submit runnable codes and a report about all details of your algorithm, including data preprocessing (normalization? fill missing? etc.), algorithm implementation, parameter setting…
- Give a presentation.

Project – Time Series Anomaly Detection

- Leaderboard Scoring rule:
  - The first place with best F-score will get 50 points.
  - Other teams: score = $\frac{your\ F-score}{best\ F-score} \times 50$

  For example, best F-score = 0.8, the F-score of your team is 0.4, you will get score = $\frac{0.4}{0.8} \times 50$=25 points

Time Series Anomaly Detection Competition Website



描述

The testing set is for ANM (Advanced Network Management) 2019 Fall Project (Anomaly Detection Challenge).

Table 1. The example of testing set.

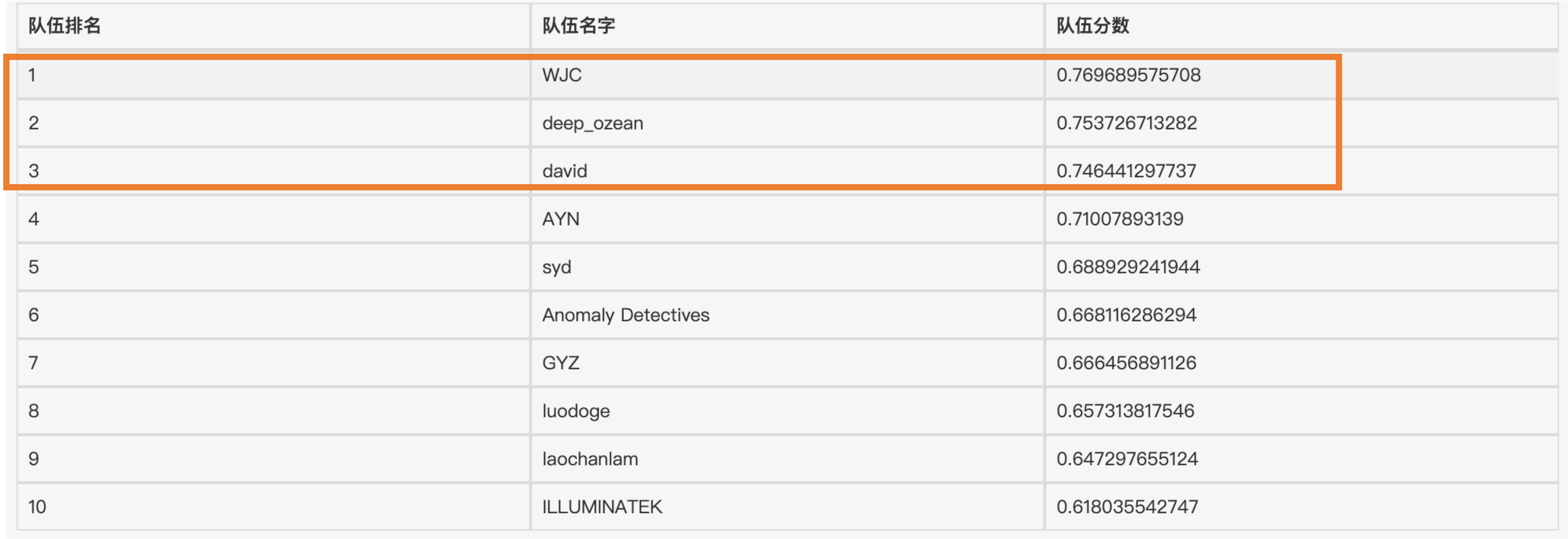
KPI ID	timestamp	value
A	1503831000	10.8
A	1503831060	12.6
...	...	...

KPI ID: denotes the name of different KPIs

Notice that there are some missing points in the dataset.

http://iops.ai/competition_detail/?competition_id=11&flag=1

Review: ANM 2018 Fall



队伍排名	队伍名字	队伍分数
1	WJC	0.769689575708
2	deep_ozean	0.753726713282
3	david	0.746441297737
4	AYN	0.71007893139
5	syd	0.688929241944
6	Anomaly Detectives	0.668116286294
7	GYZ	0.666456891126
8	luodoge	0.657313817546
9	laochanlam	0.647297655124
10	ILLUMINATEK	0.618035542747

EXIT
理学院

EXIT
理学院

4084

Thank you!

Assignment data:
https://www.dropbox.com/s/akef557hnla0h9v/ANM-data.zip?dl=0

TA: Nengwen Zhao
znw17@mails.tsinghua.edu.cn
Wechat: znw0714