NAIST Internship

Report by Panida Khuphiran

Topic

Compare performance in detect DDoS Attack by using SVM and Deep learning

Advisors | Assoc. Prof. Kohei Ichikawa (Software Design and Analysis Lab.)

Problem Description

Source

Software Define Network (SDN) is the idea that used in network system, but this idea had the soft security that is about detect the abnormal attack in the network system. From this case, many researcher try to use machine learning technic and deep learning technic to create the detect system. Main point in research is compare the performance between SVM technic and Deep Learning technic in create model.
In this research is focus on only DDoS attack.

Main Point

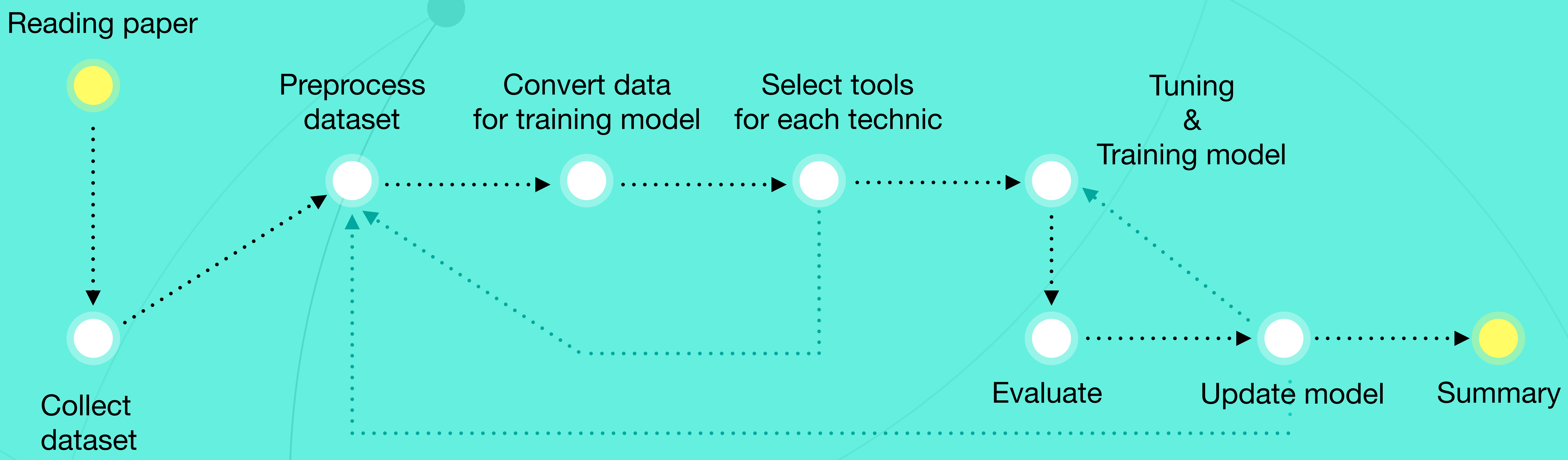
Can compare the difference results of the technic that can use this information to decide the appropriate technic for each work

Methodology

Solution

Study the feature of DDoS attack packets, and then define those feature to classify the packet into group attack or not. Next, creating the models that they are from SVM technic and Deep learning technic. In the last step, we would evaluate the model to compare the accuracy and time used and then we would summary.

Step in Research



Methodology

Step in Research

1. Reading paper

- A Deep Learning Based DDoS Detection System in Software Defined Networking (SDN)
- DDoS Detection and Analysis in SDN-based Environment Using Support Vector Machine Classifier
- Creating-Novel-Features-to-Anomaly-Network-Detection-Using-DARPA-2009-Data-set
- Deep-Learning-Approach-for-Network-Intrusion-Detection-in-Software-Defined-Networking

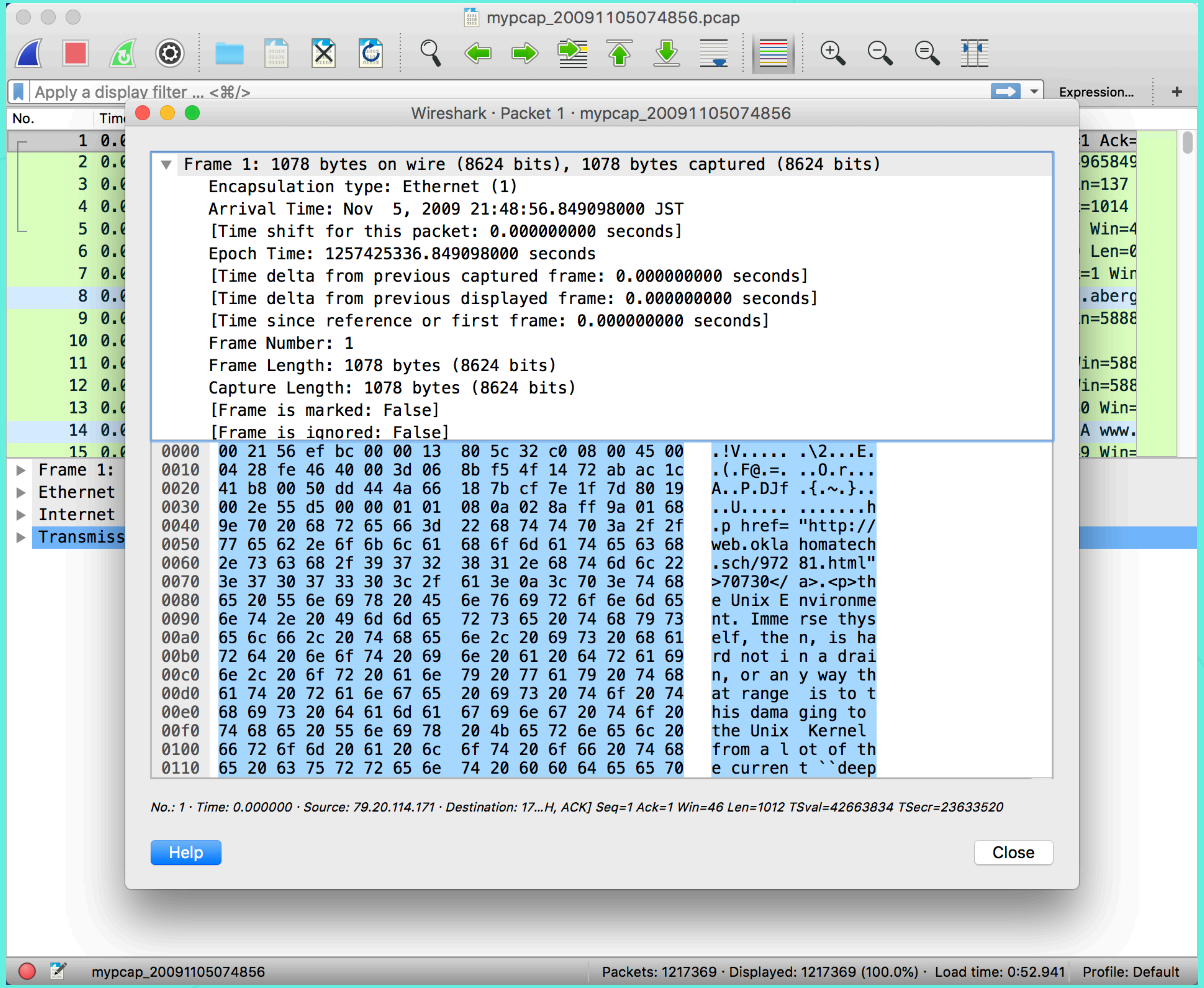
Methodology

Step in Research

2. Collect Dataset (.pcap format)

A. DARPA_2009_DDoS_attack-20091105
- using to be DDoS attack dataset

B. DARPA_Scalable_Network_Monitoring-20091103 (November 3 - 12, 2009)
- using to be normal packets dataset



mypcap_20091105074856.pcap
Apply a display filter ... <⌘/>
Expression...
No.
Time
Frame 1:
Ethernet
Internet
Transmiss
Wireshark · Packet 1 · mypcap_20091105074856
Help
Close

```
Frame 1: 1078 bytes on wire (8624 bits), 1078 bytes captured (8624 bits)
    Encapsulation type: Ethernet (1)
    Arrival Time: Nov  5, 2009 21:48:56.849098000 JST
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1257425336.849098000 seconds
    [Time delta from previous captured frame: 0.000000000 seconds]
    [Time delta from previous displayed frame: 0.000000000 seconds]
    [Time since reference or first frame: 0.000000000 seconds]
    Frame Number: 1
    Frame Length: 1078 bytes (8624 bits)
    Capture Length: 1078 bytes (8624 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
0000  00 21 56 ef bc 00 00 13  80 5c 32 c0 08 00 45 00   .!V..... .\2...E.
0010  04 28 fe 46 40 00 3d 06  8b f5 4f 14 72 ab ac 1c   .(.F@.=. ..O.r...
0020  41 b8 00 50 dd 44 4a 66  18 7b cf 7e 1f 7d 80 19   A..P.DJf .{.~.}..
0030  00 2e 55 d5 00 00 01 01  08 0a 02 8a ff 9a 01 68   ..U..... .......h
0040  9e 70 20 68 72 65 66 3d  22 68 74 74 70 3a 2f 2f   .p href= "http://
0050  77 65 62 2e 6f 6b 6c 61  68 6f 6d 61 74 65 63 68   web.okla homatech
0060  2e 73 63 68 2f 39 37 32  38 31 2e 68 74 6d 6c 22   .sch/972 81.html"
0070  3e 37 30 37 33 30 3c 2f  61 3e 0a 3c 70 3e 74 68   >70730</ a>.<p>th
0080  65 20 55 6e 69 78 20 45  6e 76 69 72 6f 6e 6d 65   e Unix E nvironme
0090  6e 74 2e 20 49 6d 6d 65  72 73 65 20 74 68 79 73   nt. Imme rse thys
00a0  65 6c 66 2c 20 74 68 65  6e 2c 20 69 73 20 68 61   elf, the n, is ha
00b0  72 64 20 6e 6f 74 20 69  6e 20 61 20 64 72 61 69   rd not i n a drai
00c0  6e 2c 20 6f 72 20 61 6e  79 20 77 61 79 20 74 68   n, or an y way th
00d0  61 74 20 72 61 6e 67 65  20 69 73 20 74 6f 20 74   at range  is to t
00e0  68 69 73 20 64 61 6d 61  67 69 6e 67 20 74 6f 20   his dama ging to 
00f0  74 68 65 20 55 6e 69 78  20 4b 65 72 6e 65 6c 20   the Unix  Kernel 
0100  66 72 6f 6d 20 61 20 6c  6f 74 20 6f 66 20 74 68   from a l ot of th
0110  65 20 63 75 72 72 65 6e  74 20 60 60 64 65 65 70   e curren t ``deep
```

No.: 1 · Time: 0.000000 · Source: 79.20.114.171 · Destination: 17...H, ACK] Seq=1 Ack=1 Win=46 Len=1012 TSval=42663834 TSecr=23633520

mypcap_20091105074856   Packets: 1217369 · Displayed: 1217369 (100.0%) · Load time: 0:52.941   Profile: Default

.pcap file can open with wireshark

*all collect by University of Southern California-Information Sciences Institute

Methodology

Step in Research

3. Preprocess Dataset (.pcap —> .csv)

- preprocess type 1 | Grouping IP, MAC address, …
  Problem : Evaluate is not good
- preprocess type 2 | using one-hot technic
  Problem : MemError, because of too many difference members
  in each attribute. Try to reduce some attribute and then train,
  evaluate is going down.
- preprocess type 3 (current doing …) | add Time relation

Why have many round of preprocess ?

– find the appropriate data after preprocess that would get more accuracy or get higher evaluate

Step in Research

4. Convert data for training model

for Deep Learning

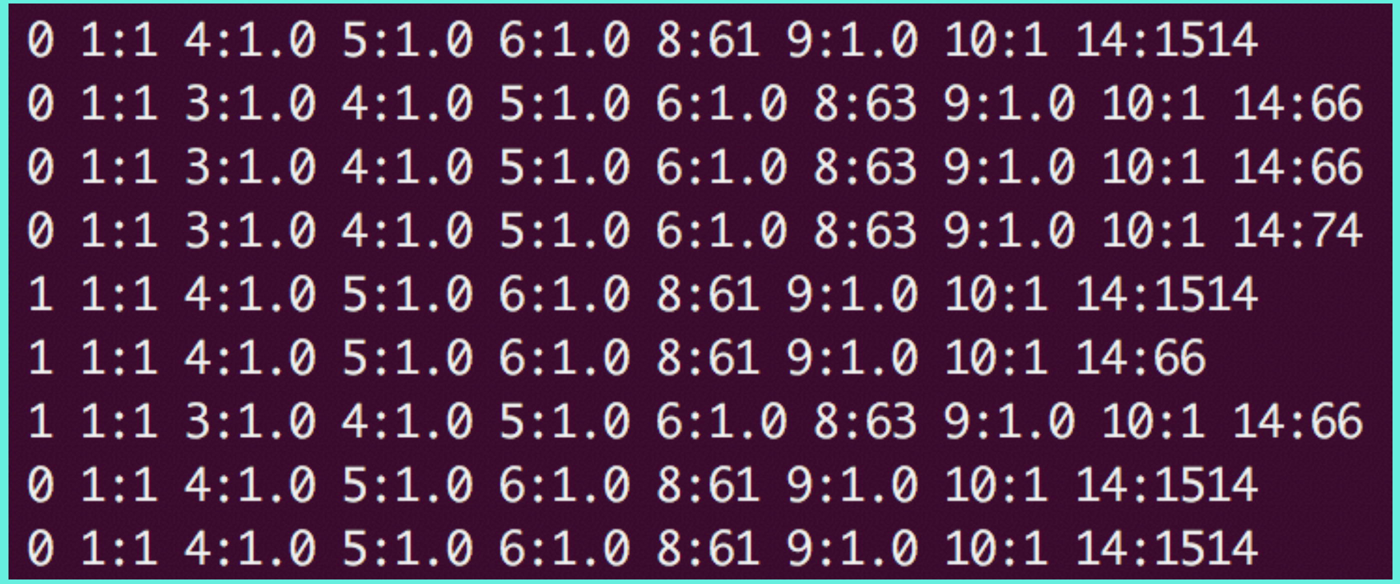
for SVM

.pcap —> dict, list —> Dataframe —> .csv —> svmlight

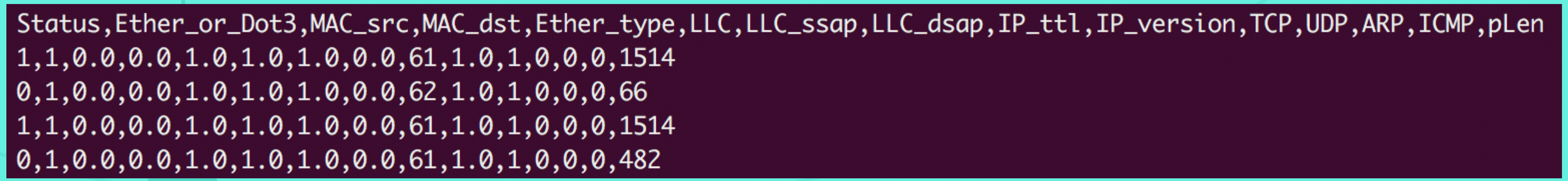
use pcapReader
from **scapy library**
(python 2.7)

pandas library
(python 2.7)

pyKMLib library
(python 2.7)



```
0 1:1 4:1.0 5:1.0 6:1.0 8:61 9:1.0 10:1 14:1514
0 1:1 3:1.0 4:1.0 5:1.0 6:1.0 8:63 9:1.0 10:1 14:66
0 1:1 3:1.0 4:1.0 5:1.0 6:1.0 8:63 9:1.0 10:1 14:66
0 1:1 3:1.0 4:1.0 5:1.0 6:1.0 8:63 9:1.0 10:1 14:74
1 1:1 4:1.0 5:1.0 6:1.0 8:61 9:1.0 10:1 14:1514
1 1:1 4:1.0 5:1.0 6:1.0 8:61 9:1.0 10:1 14:66
1 1:1 3:1.0 4:1.0 5:1.0 6:1.0 8:63 9:1.0 10:1 14:66
0 1:1 4:1.0 5:1.0 6:1.0 8:61 9:1.0 10:1 14:1514
0 1:1 4:1.0 5:1.0 6:1.0 8:61 9:1.0 10:1 14:1514
```



```
Status,Ether_or_Dot3,MAC_src,MAC_dst,Ether_type,LLC,LLC_ssap,LLC_dsap,IP_ttl,IP_version,TCP,UDP,ARP,ICMP,pLen
1,1,0.0,0.0,1.0,1.0,1.0,0.0,61,1.0,1,0,0,0,1514
0,1,0.0,0.0,1.0,1.0,1.0,0.0,62,1.0,1,0,0,0,66
1,1,0.0,0.0,1.0,1.0,1.0,0.0,61,1.0,1,0,0,0,1514
0,1,0.0,0.0,1.0,1.0,1.0,0.0,61,1.0,1,0,0,0,482
```

Step in Research

5. Select tools for each technic

- Deep Learning | Keras using Tensorflow backend, through nvidia-docker (GPU)
- SVM | pyKMLib (GPU)

*All tools use python language

Step in Research

6. Tuning & Training model

Setting each model use
- 100,000 packets for training model (from traindata-verx.csv)
- 100,000 packets for testing model (from testdata-verx.csv)

*mix data with attack packets and normal packets

SVM Tuning

Regularisation (C) : avoid misclassifying
Selection kernel : linear kernel, polynomial kernel, exponential kernel
Gamma
Margin

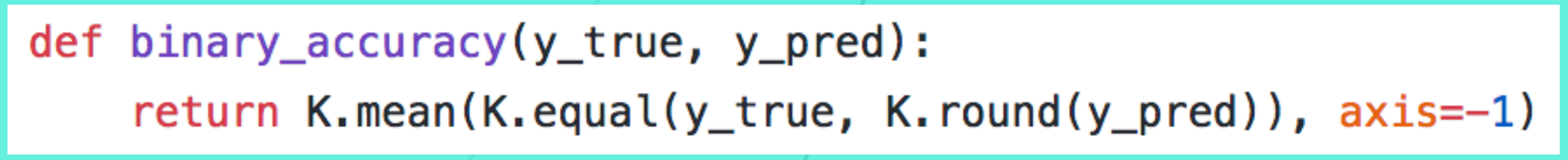
Deep learning Tuning

add layers : softmax, sigmoid, …
optimizer : sgd
loss : mse

Step in Research

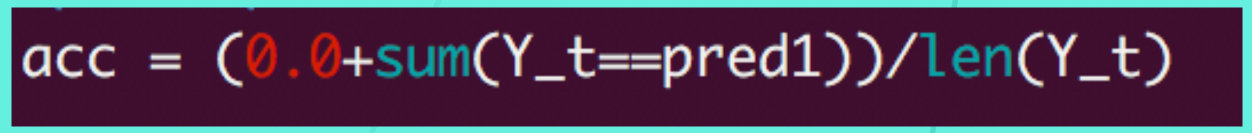
7. Evaluate

Accuracy



```python
def binary_accuracy(y_true, y_pred):
    return K.mean(K.equal(y_true, K.round(y_pred)), axis=-1)
```

Ref: https://github.com/keras-team/keras/blob/master/keras/metrics.py



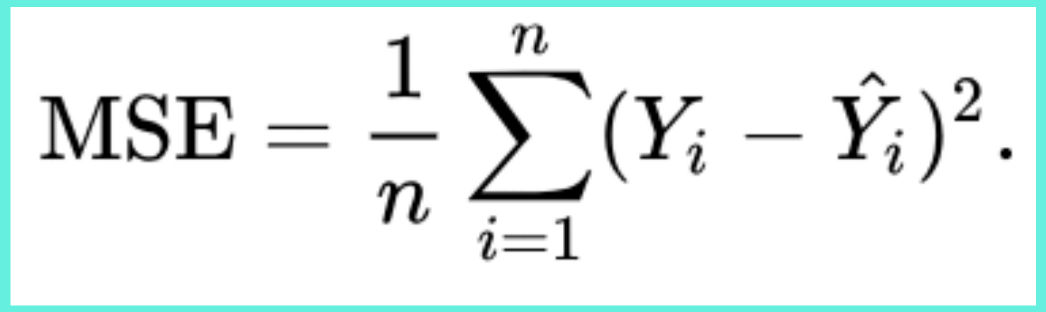
```python
acc = (0.0+sum(Y_t==pred1))/len(Y_t)
```

From | pyKMLib code

Time used

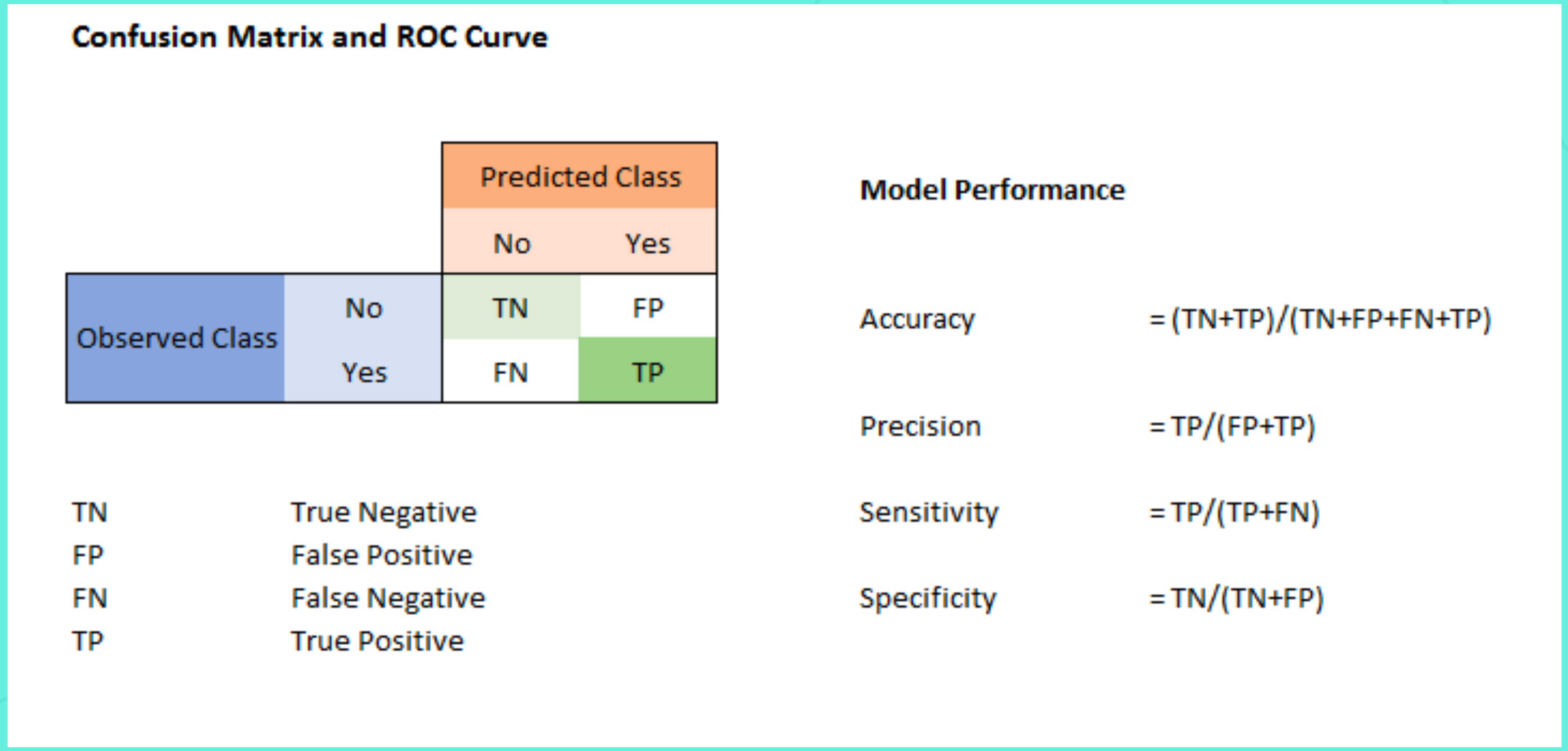
- Training model
- Predict model

Mean Square Error (MSE)



$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2.$$

Confusion Metric



Confusion Matrix and ROC Curve

		Predicted Class	
		No	Yes
Observed Class	No	TN	FP
	Yes	FN	TP

TN	True Negative
FP	False Positive
FN	False Negative
TP	True Positive

Model Performance

Accuracy	= (TN+TP)/(TN+FP+FN+TP)
Precision	= TP/(FP+TP)
Sensitivity	= TP/(TP+FN)
Specificity	= TN/(TN+FP)

Ref :
http://scaryscientist.blogspot.com/2016/03/confusion-matrix.html

Step in Research

8. Update model

Tuning model until get the highest accuracy the determine, about 60%

9. Summary

Summary the result of the evaluate model that the information is useful or appropriate for work specialty.

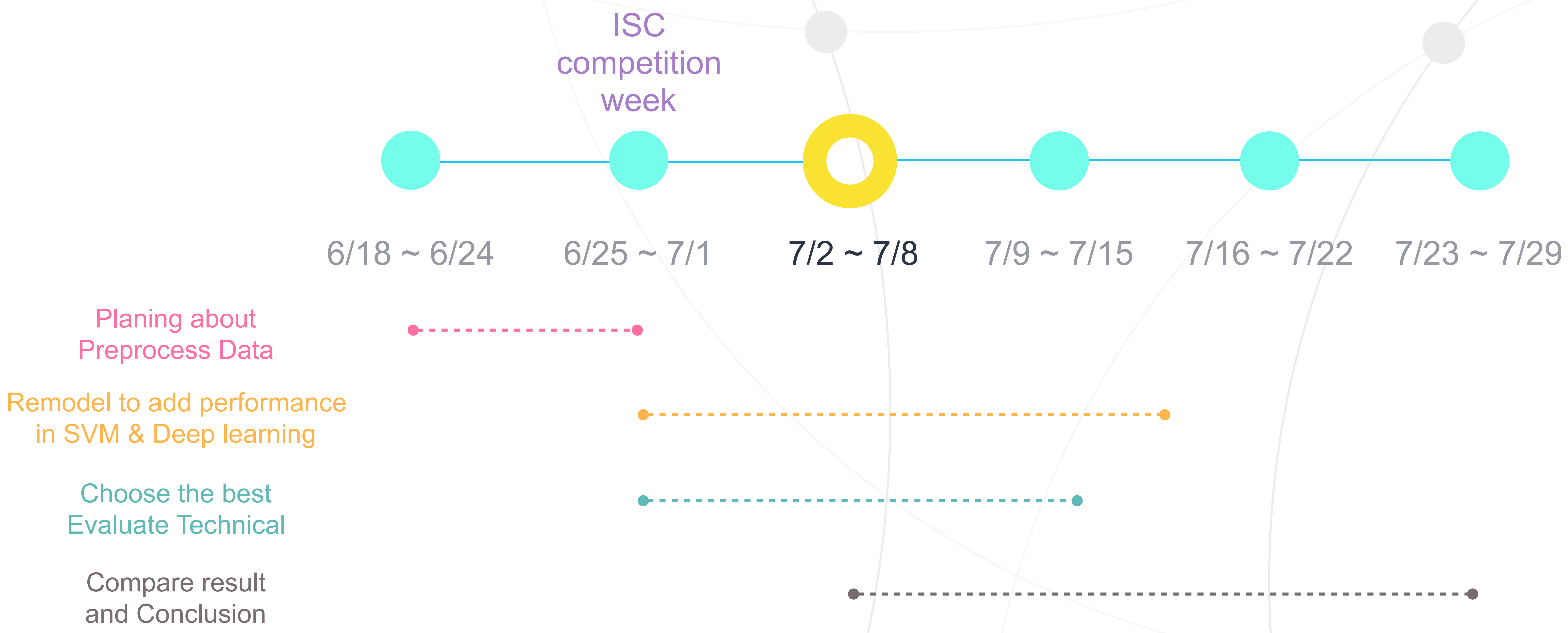
Current Progress

Current work : coding Preprocess data 3

Problem : may be have mix packets between normal packets
and attack packets in DDoS Attack dataset

Todo next : Evaluate model,
compare the result with another preprocess dataset

Plan & Todo list



ISC competition week

6/18 ~ 6/24

6/25 ~ 7/1

7/2 ~ 7/8

7/9 ~ 7/15

7/16 ~ 7/22

7/23 ~ 7/29

Planing about Preprocess Data

Remodel to add performance in SVM & Deep learning

Choose the best Evaluate Technical

Compare result and Conclusion