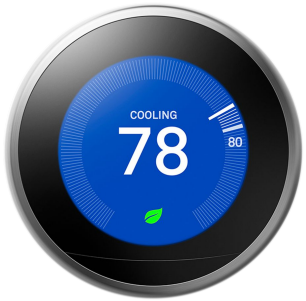
Intrusion Detection System for IoT Devices

Md Jakaria, Tasmia Shahriar, Marcus Shenker-Tauris, Nihit Mittal
{mjakari, tshahri, mshenke, nmittal2}@ncsu.edu
CSC 522: Automated Learning and Data Analysis
Project Group: P16
Date: 12/04/2023

Introduction

Intrusion detection system: An intrusion detection system (IDS) is a security mechanism designed to monitor and identify unauthorized or suspicious activities within a computer network[1].

IoT Devices: The Internet of things describes devices with sensors, processing ability, software and other technologies that connect and exchange data with other devices and systems over the Internet or other communications networks[2].



1. https://www.geeksforgeeks.org/intrusion-detection-system-ids/
2. https://en.wikipedia.org/wiki/Internet_of_things

Introduction (cont.)

BoT-IoT Dataset:

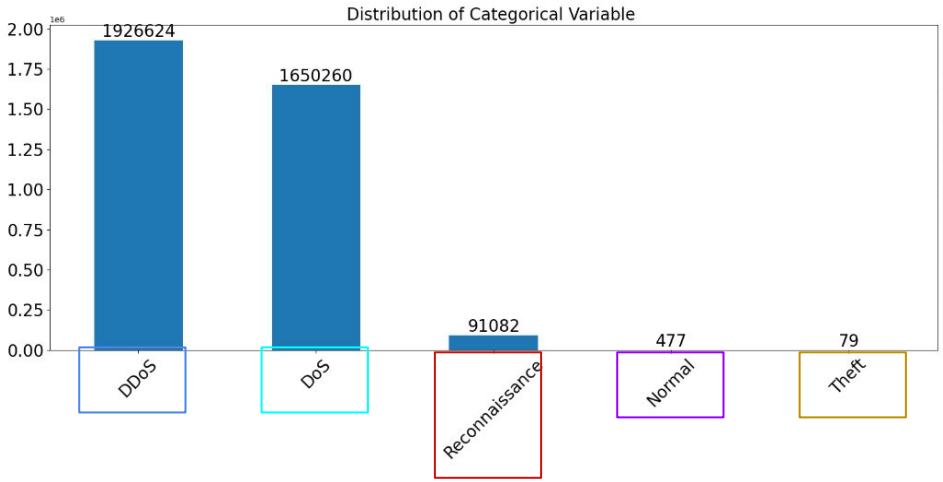
- Developed by ACCS Cyber Range Lab:
  - Simulates a network of IoT services with temperature, pressure, and humidity sensors.
  - Generates a mix of normal and botnet traffic records
  - Has 72M records and 43 features
  - Labels:

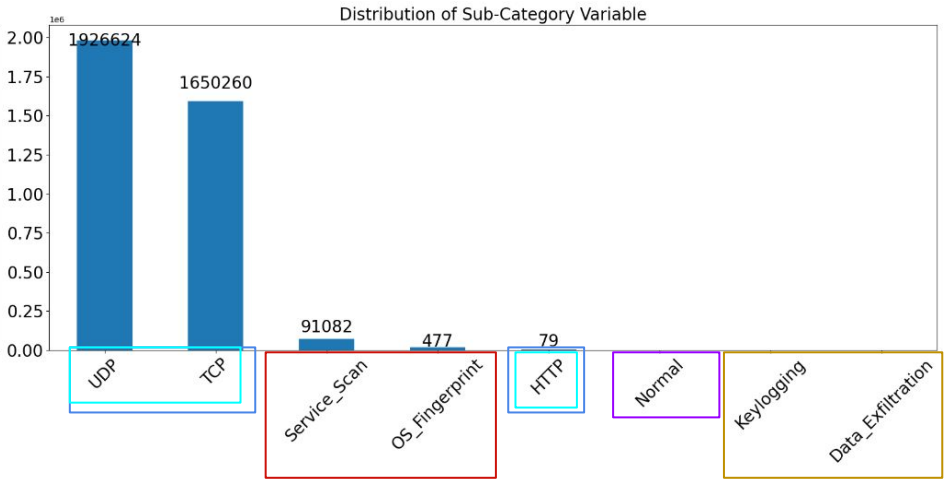
Category	Sub-category
DoS	TCP, UDP, HTTP
DDos	TCP, UDP, HTTP
Theft	Keylogging, Data-Exfiltration
Reconnaissance	Service-Scan, OS-Fingerprint
Normal	Normal

Task: Given a network traffic record, identify the category and the sub-category.

Data Preprocessing

Initial data distribution for the target variables is highly imbalanced.



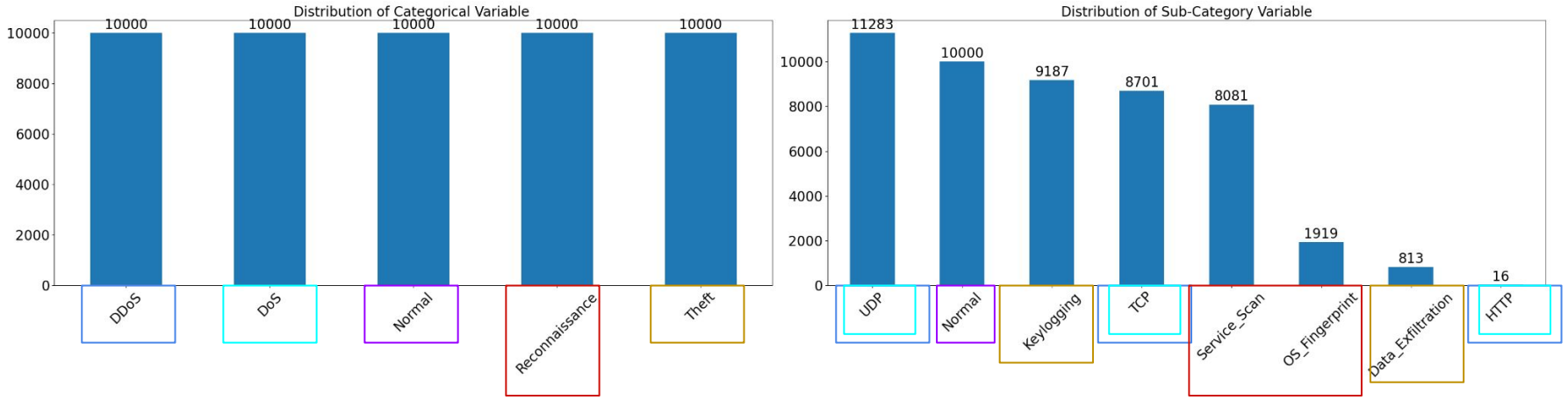


Distribution of different target classes

Data Preprocessing (cont.)

To tackle the problem of **Data Distribution**, we tried two things:

1. Gathered the data from the complete data set
2. Did an undersampling and oversampling



Data Distribution after processing

Data Preprocessing (cont.)

Pros

1. Reduced Bias
2. Reduced Resource Requirements
3. Feasibility
4. Prevent Overfitting

Cons

1. Potential Error (Information Loss)
2. Population Heterogeneity

Feature selection

- Our dataset contains 43 features:
  - Nominal: 8
  - Ratio: 32
- We used information gain to select the best nominal features.

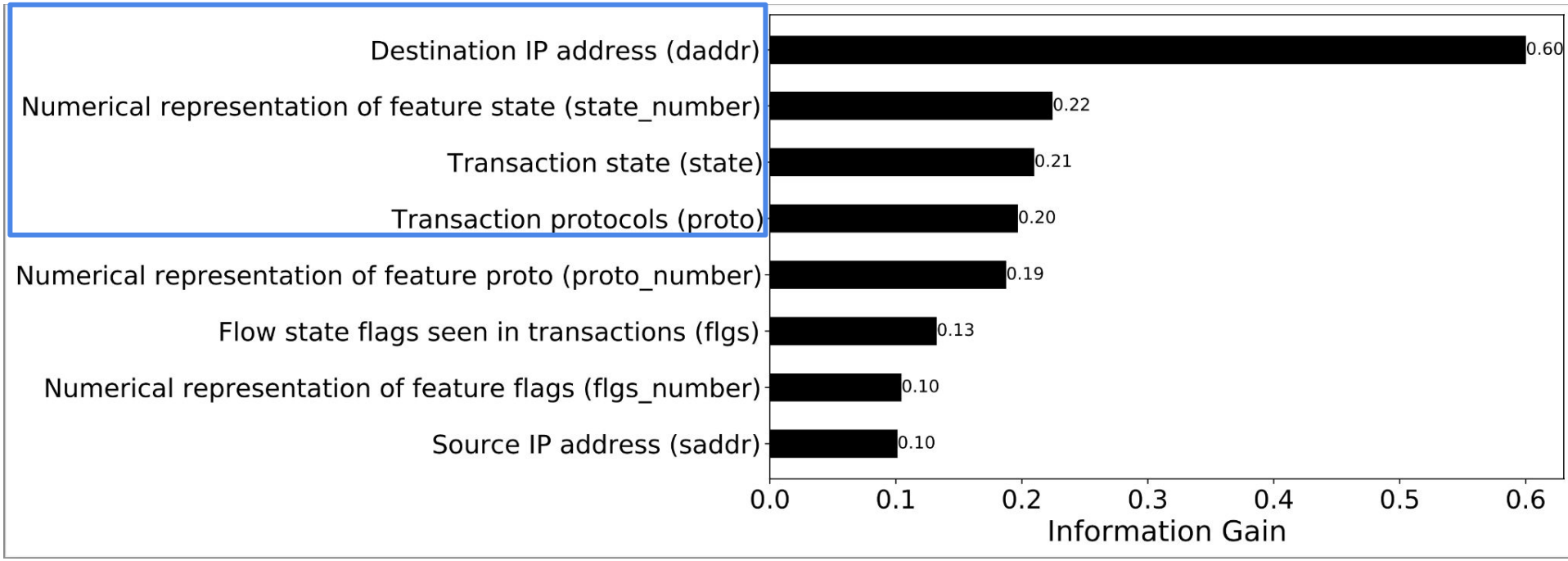
$$IG(Y|X)^* = H(Y)-H(Y|X)$$

  - H(Y|X) is uncertainty of label Y when X is known.
  - H(Y) is the uncertainty of label Y.

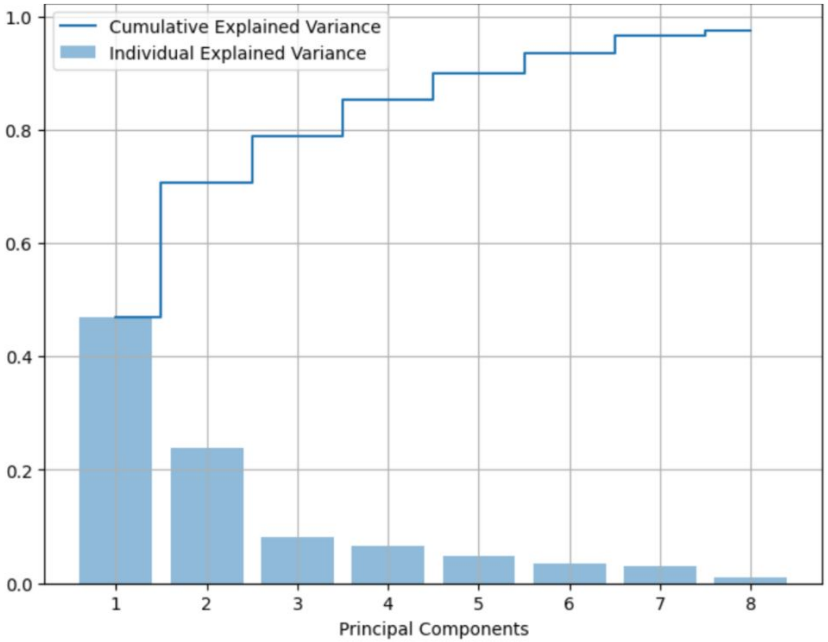
$$H(X) = \sum_i - p(X = i) \log_2 p(X = i)$$

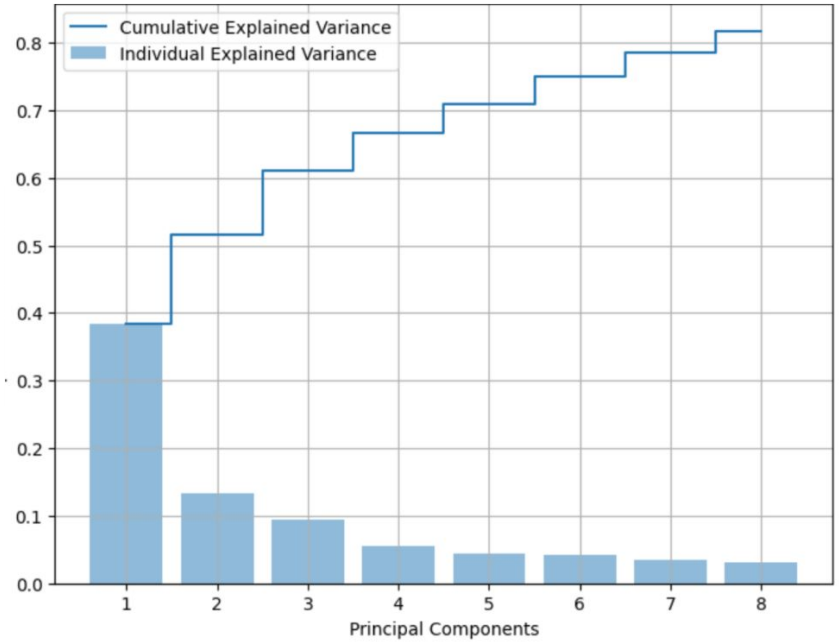
- We used Principal Component Analysis to select the best numerical features.

Feature selection (cont.)



Feature selection (cont.)





Normalized Dataset

Standardized Dataset

Explained variance by the PCA components

Feature selection (cont.)

Features	PCA1	PCA2	PCA3	PCA4	PCA5	PCA6	PCA7	PCA8
max	**0.50**	**0.43**	0.15	-0.08	-0.18	0.02	-0.04	0.00
mean	**0.38**	0.41	-0.14	-0.09	-0.06	0.01	-0.05	0.03
stddev	0.29	0.05	**0.66**	0.01	-0.28	0.04	0.02	-0.05
min	0.17	**0.33**	**-0.59**	-0.22	0.16	0.00	-0.10	-0.02
seq	0.30	-0.07	0.26	-0.05	**0.86**	-0.22	0.20	0.00
N_IN_Conn_P_DstIP	**0.48**	**-0.47**	-0.21	0.24	0.04	**0.66**	0.00	0.03
N_IN_Conn_P_SrcIP	**0.39**	**-0.43**	-0.18	0.16	-0.21	**-0.68**	**-0.31**	0.04
sum	0.01	0.11	-0.05	0.30	-0.01	-0.04	0.12	0.16
Pkts_P_State_P_Protocol_P_DestIP	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.02
Pkts_P_State_P_Protocol_P_SrcIP	0.00	0.00	0.00	-0.01	0.01	0.00	0.01	0.02
TnP_PDstIP	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01
TnP_Per_Dport	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01
TnP_PSrcIP	0.00	0.00	0.00	0.00	0.00	0.00	0.01	0.01
TnP_PerProto	-0.14	0.19	0.16	**0.36**	0.26	0.14	**-0.82**	0.07
spkts	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
pkts	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TnBPDstIP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TnBPSrcIP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
sbytes	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
bytes	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
dbytes	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
srate	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
dpkts	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
drate	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
AR_P_Proto_P_SrcIP	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
AR_P_Proto_P_Sport	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
rate	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
AR_P_Proto_P_Dport	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
AR_P_Proto_P_DstIP	0.00	0.00	0.00	0.00	0.00	0.01	0.00	0.00
dur	-0.02	0.13	-0.02	**0.32**	-0.02	-0.06	0.23	**0.81**

Feature selection (cont.)

Nominal features (3) | Normalized Ratio features (9) | Classes

daddr	state	state_number	max	mean	stddev	min	seq	N_IN_Conn_P_SrcIP	N_IN_Conn_P_DstIP	TnP_PerProto	dur	category	subcategory
192.168.100.3	RST	1.00	0.06	0.06	0.00	0.06	0.85	1.00	1.00	0.00	0.00	DDoS	TCP
192.168.100.3	REQ	3.00	0.67	0.65	0.05	0.63	0.90	0.97	1.00	0.00	0.00	DDoS	TCP
192.168.100.3	INT	4.00	0.84	0.74	0.26	0.56	0.59	0.25	1.00	0.00	0.01	DDoS	UDP
192.168.100.6	RST	1.00	0.04	0.01	0.03	0.00	0.38	0.45	0.45	0.00	0.02	DoS	TCP
192.168.100.6	REQ	3.00	0.01	0.00	0.00	0.00	0.82	0.28	0.28	0.00	0.01	DoS	TCP
192.168.100.3	RST	1.00	0.03	0.03	0.00	0.03	0.99	1.00	1.00	0.00	0.00	DoS	TCP
192.168.100.7	REQ	3.00	0.00	0.00	0.00	0.00	0.77	1.00	1.00	0.00	0.01	DoS	TCP
224.0.0.251	INT	4.00	0.00	0.00	0.00	0.00	0.06	0.08	0.23	0.88	0.00	Normal	Normal
224.0.0.251	INT	4.00	0.00	0.00	0.00	0.00	0.06	0.11	0.23	0.88	0.00	Normal	Normal
224.0.0.251	INT	4.00	0.00	0.00	0.00	0.00	0.06	0.11	0.23	0.88	0.00	Normal	Normal
224.0.0.251	INT	4.00	0.00	0.00	0.00	0.00	0.06	0.11	0.23	0.88	0.00	Normal	Normal
224.0.0.251	INT	4.00	0.00	0.00	0.00	0.00	0.06	0.11	0.23	0.88	0.00	Normal	Normal
192.168.100.3	INT	4.00	0.00	0.00	0.00	0.00	0.03	0.13	0.58	0.00	0.00	Reconnaiss	Service_Scan
192.168.100.3	INT	4.00	0.00	0.00	0.00	0.00	0.03	0.14	0.58	0.00	0.00	Reconnaiss	Service_Scan
192.168.100.3	RST	1.00	0.00	0.00	0.00	0.00	0.00	1.00	1.00	0.00	0.00	Reconnaiss	Service_Scan
192.168.100.3	FIN	6.00	0.06	0.06	0.00	0.06	0.08	0.22	1.00	0.01	0.00	Reconnaiss	Service_Scan
192.168.100.14	URP	5.00	0.00	0.00	0.00	0.00	0.01	0.42	0.10	0.00	0.00	Reconnaiss	Service_Scan
192.168.100.14	RST	1.00	0.00	0.00	0.00	0.00	0.00	0.47	0.22	0.00	0.00	Theft	Keylogging
192.168.100.15	RST	1.00	0.00	0.00	0.00	0.00	0.00	0.47	0.24	0.00	0.00	Theft	Keylogging
192.168.100.14	RST	1.00	0.00	0.00	0.00	0.00	0.00	0.47	0.22	0.00	0.00	Theft	Keylogging
192.168.100.14	RST	1.00	0.00	0.00	0.00	0.00	0.00	0.47	0.22	0.00	0.00	Theft	Keylogging
192.168.100.3	FIN	6.00	0.10	0.10	0.00	0.10	0.00	0.18	0.07	0.00	0.00	Theft	Data_Exfiltration
192.168.100.14	RST	1.00	0.00	0.00	0.00	0.00	0.00	0.47	0.22	0.00	0.00	Theft	Keylogging

Initial Methods Development

- Train/test split: 80-20 split
- Models used (9): Decision Tree, Naive Bayes, K-Nearest Neighbors, Logistic Regression, Random Forest, Gradient Boost, Support Vector Machine, ANN, RNN
- Ran on categories: DDos, Dos, Normal, Reconnaissance, Theft
- Simple neural networks:
  - Two 32 unit hidden layers with ReLu activation
  - Softmax activation output layer
  - Epochs: 10

Initial Results

- Key measure: Recall

Model	DT	NB	KNN	LR	RF	GB	SVM	ANN	RNN
Accuracy	99.86	81.52	98.89	90.55	99.94	99.87	95.05	96.36	95.91
Precision	99.86	84.30	98.90	90.54	99.94	99.87	95.24	96.32	95.95
Recall	99.86	81.54	98.90	90.59	99.94	99.87	95.10	96.32	95.90
Time	0.27s	0.02s	0.13s	3.84s	5.96s	47.54s	12.20s	43.74s	147.44s

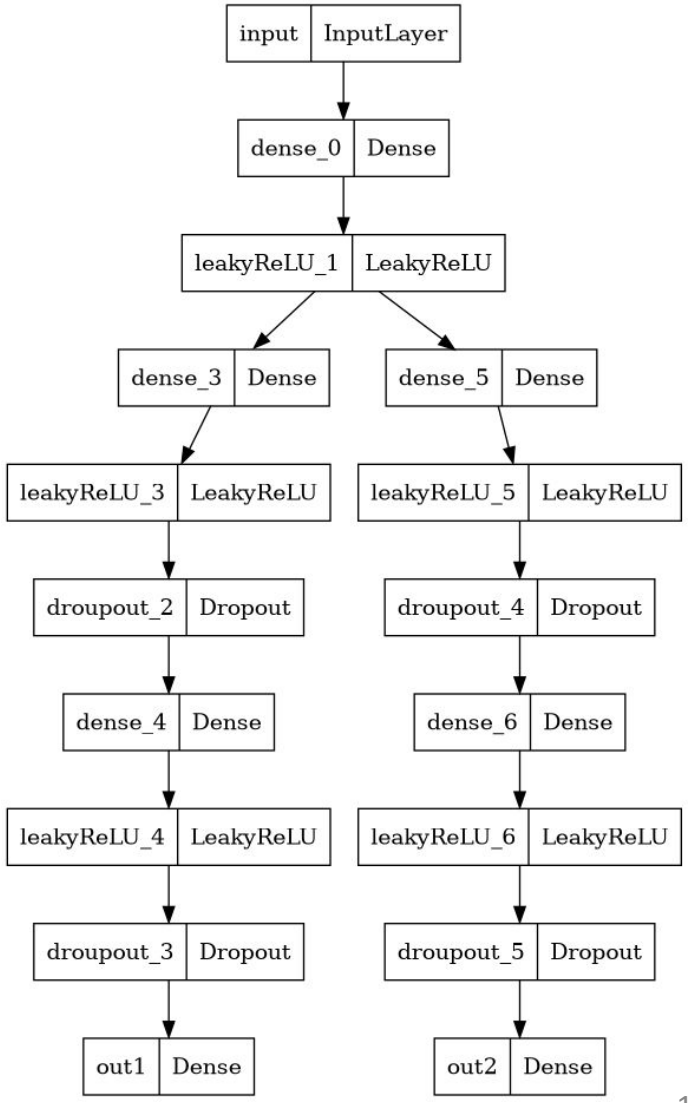
- In order of recall score:
  - Random forest, Gradient boosting, Decision tree, KNN, ANN, RNN, SVM, Logistic regression, Naive Bayes

Methods Development

Multi-Task Learning (MTL)

Layer (type)	Output Shape	Param #	Connected to
input (InputLayer)	[(None, 10)]	0	[]
dense_0 (Dense)	(None, 1024)	11264	['input[0][0]']
leakyReLU_1 (LeakyReLU)	(None, 1024)	0	['dense_0[0][0]']
dense_3 (Dense)	(None, 256)	262400	['leakyReLU_1[0][0]']
dense_5 (Dense)	(None, 256)	262400	['leakyReLU_1[0][0]']
leakyReLU_3 (LeakyReLU)	(None, 256)	0	['dense_3[0][0]']
leakyReLU_5 (LeakyReLU)	(None, 256)	0	['dense_5[0][0]']
droupout_2 (Dropout)	(None, 256)	0	['leakyReLU_3[0][0]']
droupout_4 (Dropout)	(None, 256)	0	['leakyReLU_5[0][0]']
dense_4 (Dense)	(None, 128)	32896	['droupout_2[0][0]']
dense_6 (Dense)	(None, 256)	65792	['droupout_4[0][0]']
leakyReLU_4 (LeakyReLU)	(None, 128)	0	['dense_4[0][0]']
leakyReLU_6 (LeakyReLU)	(None, 256)	0	['dense_6[0][0]']
droupout_3 (Dropout)	(None, 128)	0	['leakyReLU_4[0][0]']
droupout_5 (Dropout)	(None, 256)	0	['leakyReLU_6[0][0]']
out1 (Dense)	(None, 5)	645	['droupout_3[0][0]']
out2 (Dense)	(None, 8)	2056	['droupout_5[0][0]']

Total params: 637453 (2.43 MB)
Trainable params: 637453 (2.43 MB)
Non-trainable params: 0 (0.00 Byte)

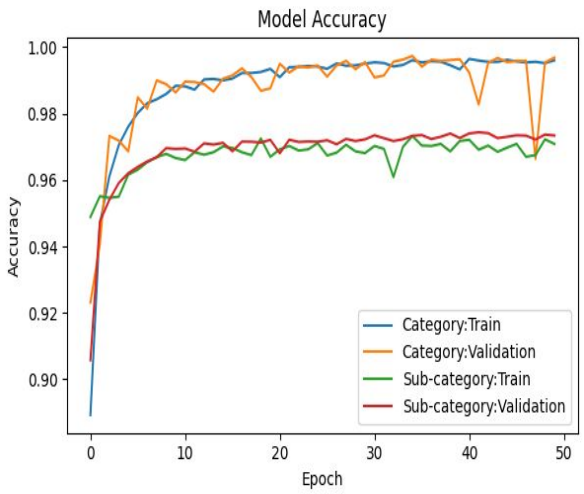


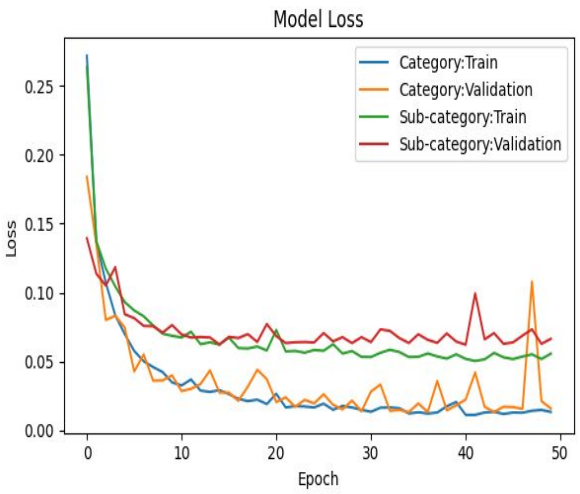
Methods Development: Performance

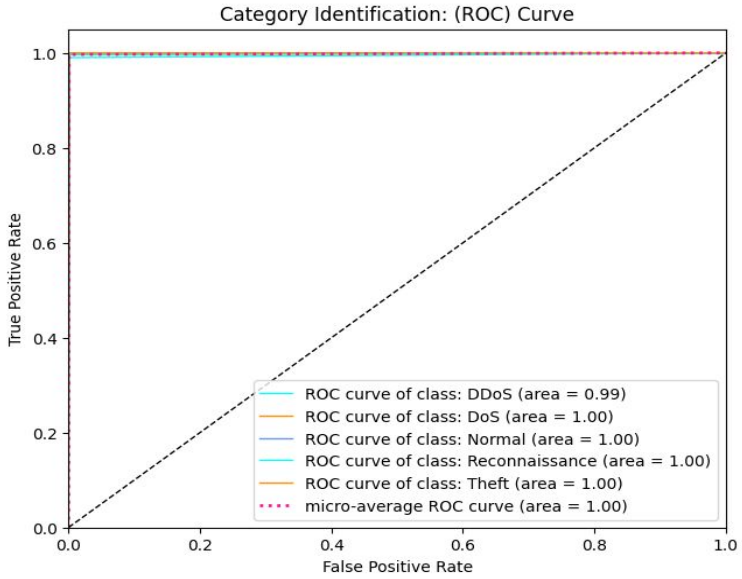
Platform: Kaggle

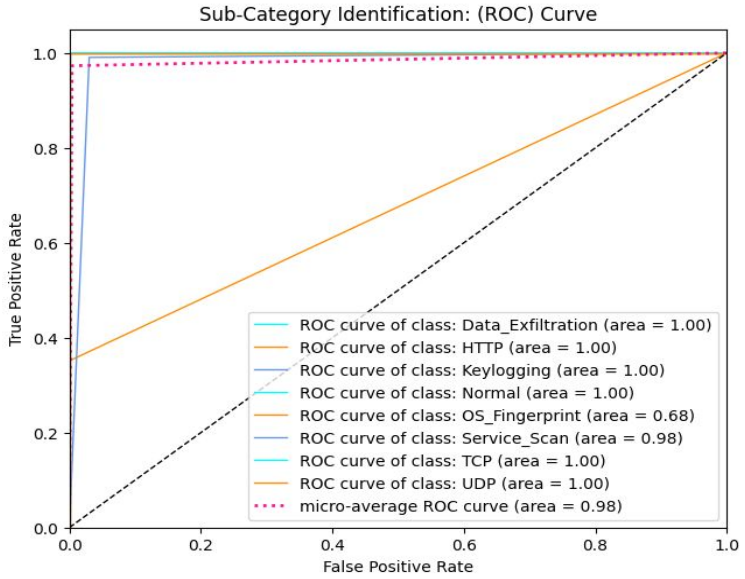
RAM:

Metric	Category	Sub-category
Accuracy	0.9887	0.9734
Precision	0.9888	0.9715
Recall	0.9887	0.9734
F1 Score	0.9886	0.971









Discussion / Conclusion

- Random Forest yield the highest accuracy and recall for the category prediction task.
- We also built a multitask model to predict both category and sub-category.
- The multitask model yield 98.87 recall in category prediction task and 97.34 in the sub-category prediction task.
- Performance for sub-category identification was less due to imbalance sample size for sub-categories.