Kitsune Network Attack Classification Using Machine Learning

Kyungbin Lee, Nahid Ebrahimi Majd

Department of Computer Science and Information System

California State University, San Marcos, United States

ABSTRACT

The rise of network attacks has emerged as a pressing concern for the technology industry. Either an IP-based commercial surveillance system or a network full of IoT devices are required to identify new types of network attacks or utilize a dynamic analysis. The term "Kitsune" is known for the intelligence and trickery of fox spirits in Japanese folklore. We performed filter methods of feature selection and then applied Decision Tree(DT), Random Forest(RF), Naïve Bayes (NB), Logistic Regression(LR), Support Vector Machine(SVM), k-Nearest Neighbor (KNN), Extreme Gradient Boost(XGB), Gradient Boosting (GB) and Extra Trees (ET) on a Kitsune network Attack dataset that contains 9 types network attack. The experimental results demonstrate that RF classifier with ANOVA feature selection outperforms either in binary classification or in multi-label(family) classification than other ML models.

OBJECTIVE

The number of network attacks on computer system has been increasing over the years. In the technology industry, network attacks on businesses have become an increasing concern. It is important to build the system to detect various types of network attacks. A successful network attack can have serious implications, including the loss of crucial data, financial losses, and legal liability. Kitsune network attack dataset is a collection of the traffic of 9 types of network attacks captured from either an IP-based commercial surveillance system or a network full of IoT devices. This research investigates the efficiency of various ML models to predict network attack in binary classification and multi-label classification as Figure 2 & 3.

Attack Type	Attack Name	Tool	Description: *The attacker…*	Violation	Vector	# Packets	Time [min.]
Recon.	OS Scan	Nmap	*…scans the network for hosts, and their operating systems, to reveal possible vulnerabilities.*	C	1	1,697,851	52.2
	Fuzzing	SFuzz	*…searches for vulnerabilities in the camera's web servers by sending random commands to their cgis.*	C	3	2,244,139	85.5
Man in the Middle	Video Injection	Video Jack	*…injects a recorded video clip into a live video stream.*	C, I	1	2,472,401	33.4
	ARP MitM	Ettercap	*…intercepts all LAN traffic via an ARP poisoning attack.*	C	1	2,504,267	28.2
	Active Wiretap	Raspberry PI 3B	*…intercepts all LAN traffic via active wiretap (network bridge) covertly installed on an exposed cable.*	C	2	4,554,925	95.6
Denial of Service	SSDP Flood	Saddam	*…overloads the DVR by causing cameras to spam the server with UPnP advertisements.*	A	1	4,077,266	40.8
	SYN DoS	Hping3	*…disables a camera's video stream by overloading its web server.*	A	1	2,771,276	52.8
	SSL Renegotiation	THC	*…disables a camera's video stream by sending many SSL renegotiation packets to the camera.*	A	1	6,084,492	65.6
Botnet Malware	Mirai	Telnet	*…infects IoT with the Mirai malware by exploiting default credentials, and then scans for new vulnerable victims network.*	C, I	X	764,137	118.9

Figure 1: Kitsune network attack dataset.

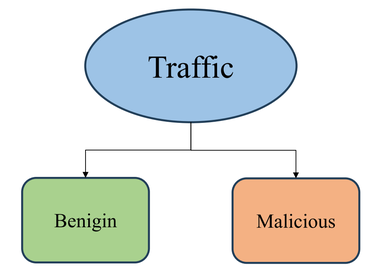

Figure 2: Binary classification in Kitsune.

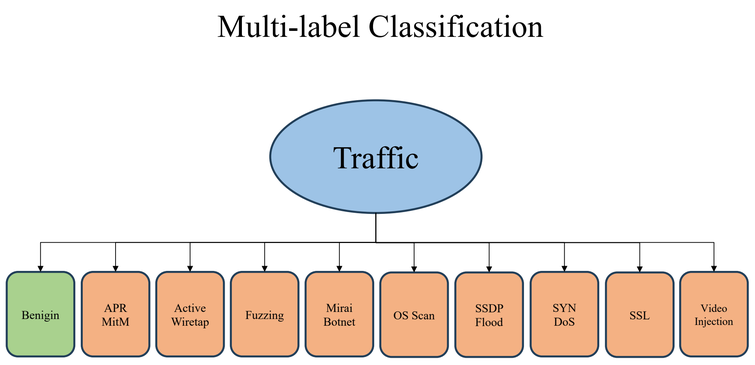

Figure 3: Multi-label classification in Kitsune.

METHODS

- Dataset

Since each dataset contains millions of network packets, we extract a subset dataset and make samples as Figure 4. Each Attack includes 6,500 benign and 6,500 malicious rows. In the training set, there are total 117,000 rows.

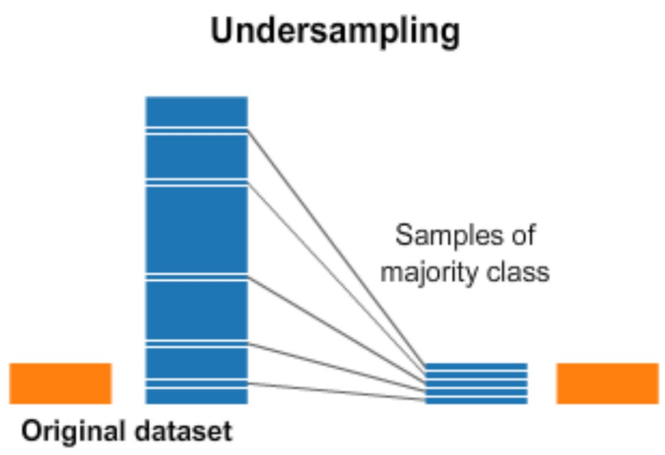

Figure 4: Undersampling method in machine learning.

- Data preprocessing

Data Split: We split the dataset into training data and test data in the ratio of 70:30, using 5-fold validation to generalize the models.

Standardization: Normalization technique is used to convert each of the variables into a similar scale by centering each variable at zero with a standard deviation of 1.

- Feature Selection Techniques

We use ANOVA(Analysis of Variance F-value) to calculate the F-value for each feature by comparing the variance of the target variable explained by the feature. There are 115 features in the dataset. In Table 1, we list the feature's number, name, and score from highest to lowest score.

Feature	Name	Score
Feature 49	HH_L1_radius	13502.728850
Feature 42	HH_L3_radius	13496.757285
Feature 102	HpHp_L0.1_weight	13470.305291
Feature 35	HH_L5_radius	13408.138259
Feature 40	HH_L3_std	12821.612847
Feature 47	HH_L1_std	12805.166998
Feature 33	HH_L5_std	12764.105476
Feature 56	HH_L0.1_radius	12556.895291
Feature 95	HpHp_L1_weight	12313.862233
Feature 54	HH_L0.1_std	11892.144859

Table 1: Top 10 Features F-values applying ANOVA F-test feature selection.

Stream aggregation	H	Traffic from packet's host (IP)
	MI	Traffic from packet's host (IP + MAC)
	HH	Traffic going from packet's host to destination host (IP)
	HH_jit	Jitter of the HH
	HpHp	Traffic going from packet's host to destination host & port
Time frame	L	How much recent history of the stream is capture
Statistics	weight	Weight of the stream
	mean	Mean of the stream
	std	Standard deviation of the stream
	radius	Root squared sum of the two streams' variances
	magnitude	Root squared sum of the two streams' means
	cov	Approximated covariance between two streams
	pcc	Approximated correlation coefficient between two streams

Table 2: Description of feature name.

- GridSearchCV

We use the hyperparameter from GridSearchCV method to get higher accuracy for each ML model. We describe 9 supervised ML algorithms with their hyperparameters in Table 3.

ML algorithm	Hyperparameter
LG	C=100, penalty='none', solver='newton-cg'
DT	criterion='entropy', max_depth=20, min_sample_leaf=9
RF	criterion='entropy', max_depth=20, n_estimators=90
GB	learning_rate=0.1, n_estimators=90
SVM	C=1000, kernel='rbf', gamma=3.0
KNN	n_neighbors=6
XGB	learning_rate=0.1, n_estimators=90
NB	priors='none', var_smoothing=0.1
ET	criterion='entropy', max_depth=90, n_estimators=30

Table 3: ML models with their hyperparameters.

RESULTS

We train each model with the different number of features with ANOVA feature selection method. RF shows the highest AUC score and accuracy in Figure 5 & 6. We describe the confusion matrix of RF for both classifications in Figure 7.

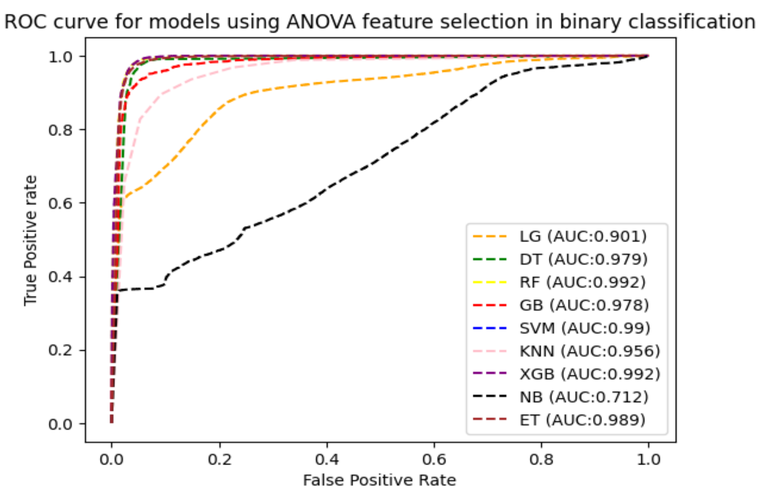

Figure 5: ROC curve for binary classification.

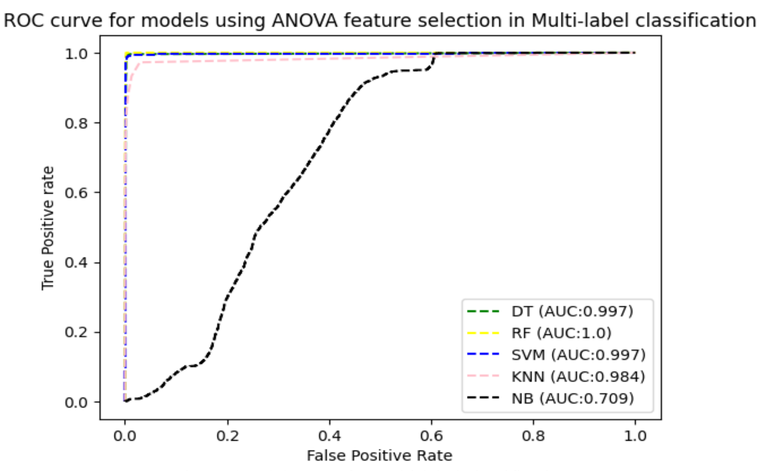

Figure 6: ROC curve for multi-label classification.

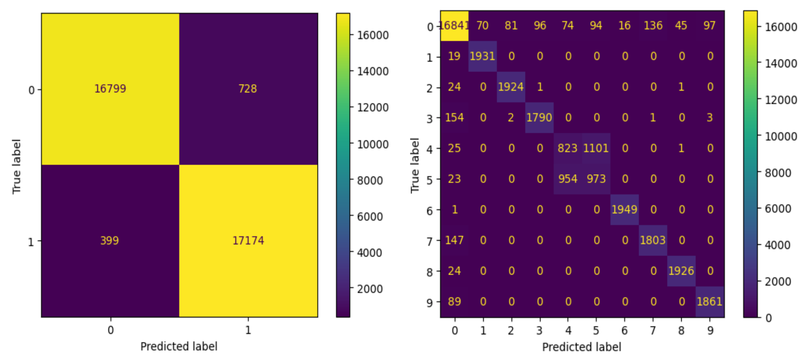

Figure 7: Confusion Matrix of RF

CONCLUSION

With the increasing threat of various types of network attacks, it is important to detect known and new forms of network attacks effectively. In this research, we designed ML models and performed extensive experiments by filtering method, feature selection method, and ML algorithms along with hyperparameter. The experimental results demonstrate that RF with ANOVA feature selection outperforms.

REFERENCE

[1] Kim, YeaSul, YeEun Kim, & Hwankuk Kim. "A Comparison Experiment of Binary Classification for Detecting the GTP Encapsulated IoT DDoS Traffics in 5G Network." *Journal of Internet Technology* [Online], 23.5 (2022): 1049-1060. Web. 12 Jul. 2023

[2] Y.Mirsky, T. Doitshman, Y.Elovici, and A. Shabtai, "Kitsune: An Ensemble of Autoencoders for Online Network Intrusion Detection," arXiv.org, May 27, 2018. http://arxiv.org/abs/1802.09089 13 Jul. 2023

[3] Kitsune Network Attack Dataset. (2019). UCI Machine Learning Repository. https://doi.org/10.24432/C5D90Q.

ACKNOWLEDGEMENT

This research was funded by grants from the following agency:

San Diego Foundation