UCI ML Hackathon

Team Name: LADZ

Dataset: DNS

Team Member: Zhen Chen, zhenc4@uci.edu
Ameer Hussain, AAHussai@uci.edu

DGA Detection and Cluster

- Anti-virus and prevent the virus from contacting the command center
- Find a way to detect the DGA (Domain Generate Algorithm) domains
- Cluster the DGA domains into families

Overview

Part 1: classify DGA

1. supervised learning
2. deep neural networks, stack models

part 2: cluster DGA

1. unsupervised learning
2. group activity features, IP features
3. domain name string features

Part 1

- Prepare the data: train 80%, validation 10%, test 10% (similar distribution)
- Benign domain name filter: top 1 million websites on Alexa https://www.alexa.com/topsites
- Models
  - Basic model: only embedding, tf-idf plus Bayes model
  - Modified MIT Model, End Game Model, Invincea Model, NYU Model
  - Xgboost stack model

Metric

~82% benign domains

~18% DGA domains

- Precision
- Recall
- Score 1 (doc_2)

Part 1: accuracy of DGA domains.
The grading formula is:

$$Score_1 = max\left(\frac{1}{N} \cdot w \cdot \sum_{k=1}^{m} ([D_k \in FTrue]), 0\right)$$

In the above formula:
FTrue: the set of the standard answer.
N: the size of the set of the standard answer (FTrue).
m: the length of the submitted domain list
D_k: the k-th domain in the submitted domain list
[]: inside is a judgment statement. If the statement holds, its value is 1. Otherwise, it's -1.
max(x,y): maximum function.
w: Constant weight.

Results without filter

	val_precision	test_precison	val_recall	test_recal	val_socre1	test_score1
Baseline (embedding)	0.7921	0.7706	0.8743	0.8477	0.6449	0.5953
modified MIT	0.8094	0.7845	0.9188	0.9290	0.7024	0.6738
End game	0.8024	0.7816	0.9250	0.9131	0.6972	0.6579
Invincea	0.7991	0.7768	0.9127	0.9009	0.6832	0.6421
NYU	0.7997	0.7788	0.9337	0.9411	0.6998	0.6738
Xgboost Stack	0.7874	0.7753	0.9634	0.9728	0.7033	0.6916

Threshold is chosen based on false positive rate = 0.05

Results

- Benign domain name filter gives about 3% improvement on score 1.
- Final results

	val_precision	test_precison	val_recall	test_recal	val_socre1	test_score1
Xgboost Stack	0.8193	0.8099	0.9337	0.9477	0.7277	0.7252

Part 2

Features (31 total features)

- Domain name features
  - 1 gram score, 2 gram score
  - Entropy 1 gram, Entropy 2 gram
  - Meaningful Character Ratio
  - DNS Length
  - Score 1
  - TLDs
- Group Activities
  - Mapping host IP address to malicious DGAs they visited and tracking families

Meaningful Character Ratio

MCR FUNCTION

For example:

If d = stackoverflow then

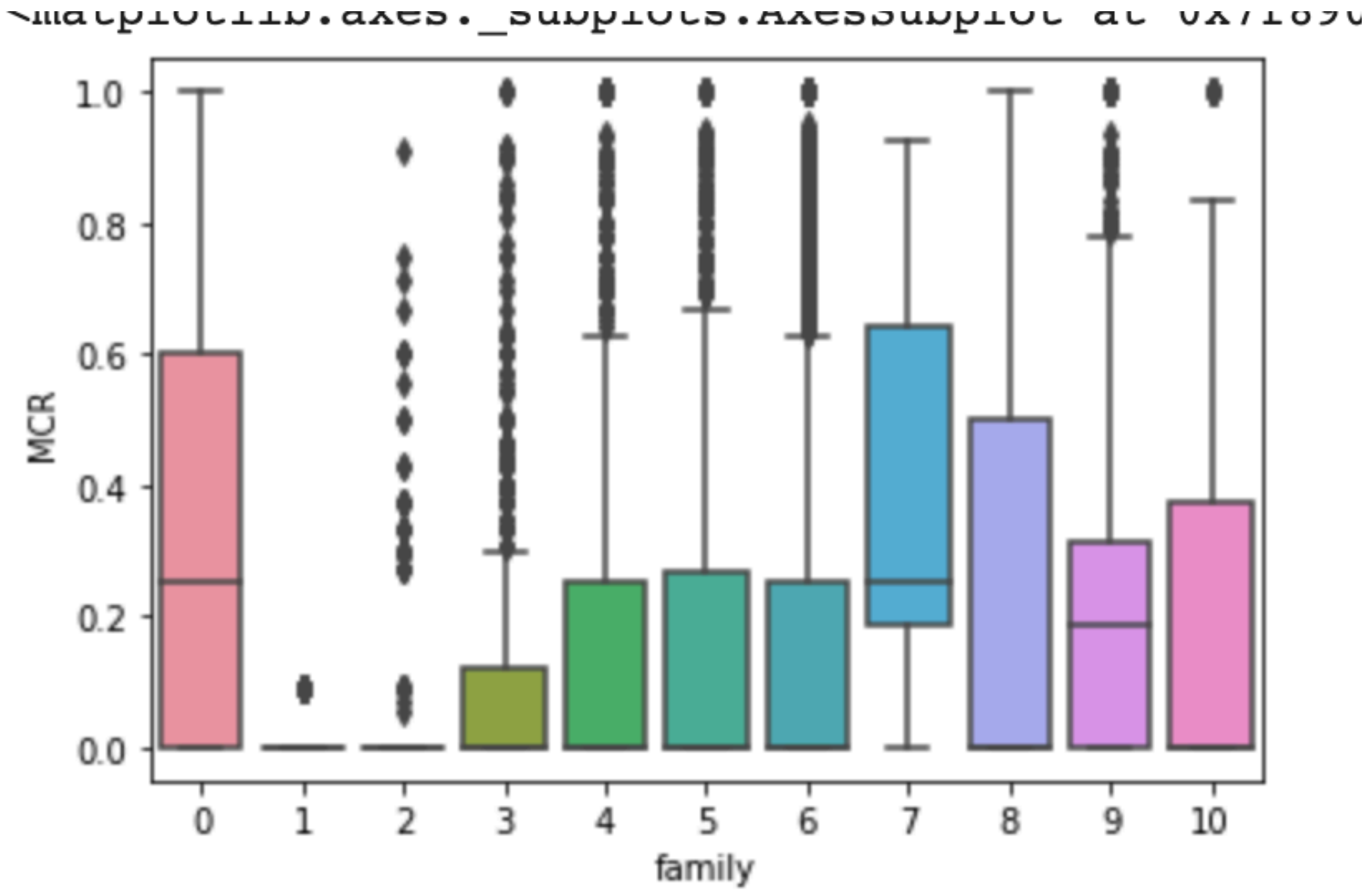
$$R(d) = \frac{|\text{stack}|+|\text{over}|+|\text{flow}|}{13} = 1.$$

If d = sixabcd then

$$R(d) = \frac{|\text{six}|}{7} = 0.4285.$$

Collapse 25 child cells under

MCR



Score 1

- How likely the model from part 1 is to classify each family as benign or malicious.

family	
1	0.987417
2	0.874736
3	0.878302
4	0.870607
5	0.845683
6	0.705242
7	0.528044
8	0.793016
9	0.778546
10	0.630459

Group Activities/IPs

	123.114.120.72	101.28.115.158	166.111.50.21	205.215.81.221	211
z54f71fcb61e49158f3648236556b04eb1.so	0	0	0	0	
l31f5dcf7f398d2b94cd9155ee392c701b.so	0	0	0	0	
d442a48406424819e686a5d1fde7c602ea.so	0	0	0	0	
zba909e91846ce27aff7902f9228ee40ea.so	0	0	0	0	
p8f5889e854374cf8e70e91f456807f83f.so	0	0	0	0	
...	...	...	...	...	
gdwrhiivyt.cc	0	0	0	0	
hyirnrnetku.cc	0	0	0	0	
isauvvu.cc	0	0	0	0	
vddiuiygf.cc	0	0	0	0	
lcdro.cc	0	0	0	0	

11052 rows × 335 columns

Group Activities/Ips insight

```
123.114.120.72 {6: 1}
101.28.115.158 {6: 1}
166.111.50.21 {2: 1, 8: 5}
205.215.81.221 {3: 19, 5: 19, 6: 12, 8: 17}
211.68.127.2 {8: 1}
112.22.90.65 {8: 1}
171.9.72.173 {8: 1}
113.121.209.134 {6: 1}
219.131.11.66 {2: 1, 6: 1}
113.85.97.158 {8: 1}
39.90.118.154 {5: 1, 6: 2, 8: 4}
139.33.104.134 {2: 20, 3: 18, 5: 19, 6: 13, 8: 21, 10: 12}
58.48.128.15 {6: 1}
113.85.99.84 {4: 1, 8: 1}
37.143.23.6 {2: 13, 3: 31, 5: 27, 6: 26, 8: 32}
89.89.143.48 {2: 18, 3: 15, 5: 15, 6: 13, 8: 19, 10: 12}
123.114.126.143 {6: 1}
132.110.76.173 {1: 165, 2: 42}
110.85.69.156 {6: 3, 8: 2}
120.229.94.103 {8: 1}
195.96.148.131 {2: 12, 3: 17, 5: 12, 6: 16, 8: 22, 10: 21}
60.183.65.41 {6: 1}
58.37.200.156 {6: 1}
166.111.8.28 {2: 74, 3: 53, 4: 246, 5: 154, 6: 815, 7: 9, 8: 1154, 9: 109, 10: 141}
223.87.210.125 {6: 1}
61.48.211.9 {6: 1}
119.39.248.121 {6: 2, 10: 1}
```

Cluster method

- **Density-based spatial clustering of applications with noise** (**DBSCAN**)
  - DBSCAN is used because it does not specify the amount of clusters produced
- **K-Means**
  - We used silhouette score to determine the optimal number of clusters
- We found DBSCAN works best

Results using professors score formula

- Random Guess: 0.0616

  Group Activities: 0.1073

  DGA string characteristics without TLD: 0.1829

  Group Activities and DGA string characteristics without TLD 0.1816

  All Features: 0.3895

  All features without IP: 0.4696

  Using only 1 gram score, MCR, length, and TLDs: 0.5873

  Only TLD (Top Level Domain): 0.9824

Insights about TLDs and Domain String

- Weakness of our model
  - If we get a new family that uses TLD of existing family, it will misclassify
  - Low accuracy without TLDs due to the similarity in the meaningful words, ngramscores, and entropies.
  - Our model is also hindered by hosts being infected by multiple families which makes it difficult to use the IP information

family	tld	
1	in	42
	so	41
	tk	40
	to	42
2	ws	237
3	biz	1479
4	ru	868
5	info	995
6	net	3195
7	co	31
8	org	3145
9	cc	25
	online	28
	uk	551
10	cc	334

Insight about TLDs and Domain String

- We found TLD was the most important feature for clustering
- However, when we performed supervised learning without TLD for part 2, we achieved 80% accuracy with the MIT model.
- When we included TLDs as well we achieved 94% accuracy
- This shows us that it is possible to group the domains by families without the TLDs using **ONLY** string features.

Thank you for your time

- ☺