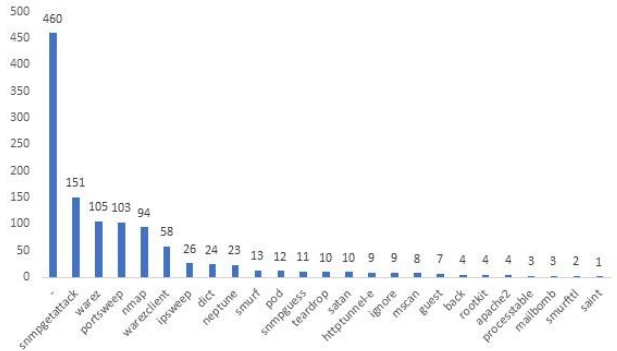
Graphs are Documents: Predicting TCP Connection Attack via Weighted Jaccard Similarity

AI607 Final Project

Minseok Choi, Sanghyeon Lee

Motivation

- Predicting TCP connection attacks can be viewed as detecting similarities or patterns of a certain type of attack.
- Document similarities are often used to detect **plagiarism**.
- Can we process graphs as documents and find document similarities of TCP connection histories?
- One simple way of computing document similarities is **Jaccard similarity**.
- But our dataset suffers from **class imbalance**.
- Can we do better?
- Apply weights!



Approach

1. Shingling

$S(D)$: *The number of the unique set in the Documents*

2. Jaccard Similarity

$$J(s_1, s_2) = \frac{|s_1 \cap s_2|}{|s_1 \cup s_2|}$$

2. Weighted Jaccard Similarity

$$J_w = J_1 + J_2 + \cdots + J_C \ , J_c = \frac{w_c * |S(D_1(c) \cap S(D_2(c)|}{\sum_{i \in C} w_i * |S(D_1(i) \cup S(D_2(i)|},$$

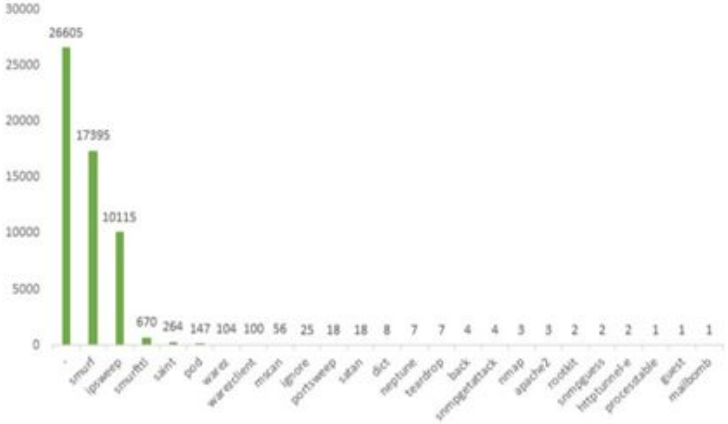


Fig. 1Number of unique 2-shingles per class

Approach

4. Pseudo F1

- We calculate the similarity for each class

$S(D)$: *The number of the unique set in the Documents*

Class	C1	C2	C3	C4	C5	C6
GT	1	0	0	1	0	0
Prediction (Similarity of each class)	0.6	0.2	0.1	0.1	0.4	0.2

$$\text{Pseudo Precision} = \frac{||\text{GT}\odot\text{Prediction}||_1}{||\text{GT}||_1}\frac{1*0.6+1*0.1}{1+1} = 0.35$$

$$\text{Pseudo Recall} = \frac{||\text{GT}\odot\text{Prediction}||_1}{||\text{Prediction}||_1} = \frac{1*0.6+1*0.1}{0.6+0.2+0.1+0.1+0.4+0.2} = 0.4375$$

$$\text{Pseudo F1} = 2 * \frac{\text{Pseudo Precision} * \text{Psudo Recall}}{\text{Pseudo Precision}+\text{Psudo Recall}} = 2 * \frac{0.35*0.4375}{0.35+0.4375} = 0.389$$

5. Optimize Pseudo F1

$$\text{Objective functnion} : \text{L} = (1 - \text{Psudo F1})^2$$

Experimental Results

Setting

1. **Build K-Shingles of all documents (train, valid, test)**
2. **Choose the threshold (score of Weighted Jaccard Similarity)**

1. Effect of threshold

Baseline F1 ('-') : 0.719

Threshold	Vanilla	Ours
0.1	0.764	0.226
0.2	0.774	0.411
0.3	**0.775**	0.647
0.4	0.765	0.766
0.5	0.763	0.779
0.6	0.756	**0.785**
0.7	0.745	0.766
0.8	0.744	0.760
0.9	0.738	0.757

2. Qualifying results in multi labels

GT	Ours
'warez', 'snmpgetattack', 'nmap'	'warez', 'snmpgetattack', 'nmap'
'snmpgetattack', 'nmap'	'snmpgetattack', 'nmap'
'smurf', 'snmpguess', 'nmap'	'smurf', 'snmpguess', 'nmap'
'warez', 'snmpgetattack', 'nmap'	'-'
'ignore', 'portsweep', 'dict', 'snmpgetattack', 'nmap', 'warez'	'ignore', 'portsweep', 'dict', 'snmpgetattack', 'nmap', 'warez'
'snmpgetattack'	'snmpgetattack', 'processtable', 'httptunnel-e', 'neptune', 'warez', 'apache2

Conclusion

- Our model using weighted Jaccard similarity **performed far better** than random guesses, as well as the vanilla Jaccard, demonstrating that graphs can be represented as a certain kind of document.
- Because our model is capable of shingling, it can **adapt to massive data** by expanding it to various algorithms, such as min-hashing and locality-sensitive hashing.
- Nevertheless, our approach did not consider the **temporality** of the data, as well as the port number, which could be important factors when determining the type of attack. Incorporating such metadata may be an interesting future work of representing graphs as documents.

References

(1) Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformer for language understanding. *arXiv preprint arXiv: 1810.04805, 2018.*

(2) Aristides Gionis, Piotr Indyk, Rajeev Motwani, et al. Similarity search in high dimensions via hashing. In *Bldb*, volume 99, pages 518-529, 1999

(3) Karen Sparck Jones, A statistical interpretation of term specificity and its application in retrieval, *Journal of documentation*, 1971

(4) Diederik P Kingma and Jimmy Ba, Adam: A method for stochastic optimization. In *ICLR*,2015.

(5) Quoc Le and Tomas Mikolov, Distributed representations of sentences and documents. In *ICML*, pages 1188-1196, 2014