MCoM: A Semi-Supervised Method for Imbalanced Tabular Security Data

Mahmoud Zamani, Xiaodi Li, Latifur Khan, Kevin Hamlen

Department of Computer Science, The University of Texas at Dallas, TX, USA

THE UNIVERSITY OF TEXAS AT DALLAS

CENTERS OF ACADEMIC EXCELLENCE IN CYBERSECURITY COMMUNITY

CAE IN CYBERSECURITY COMMUNITY

ABSTRACT

MCoM (Mixup Contrastive Mixup) is a novel semi-supervised learning approach that introduces an innovative triplet mixup data augmentation technique to tackle the imbalanced data issue in tabular security datasets. In cybersecurity, tabular datasets are notorious for their severe data imbalances, where only a few labeled attack samples exist amidst a vast sea of mostly unlabeled benign data. While semi-supervised learning has been extensively explored in image and language domains, it remains relatively underutilized in security domains, particularly when dealing with tabular security data. This domain-specific challenge involves handling complex contextual information loss and data balance issues. Experimental results involving MCoM on collected security datasets demonstrate promising outcomes, achieving state-of-the-art performance when compared to alternative methods.

OBJECTIVES

The paragraph discusses the challenging nature of software vulnerability detection, citing an example of a vulnerability in the Linux PHP interpreter (CVE 2015-3329). Despite its simplicity and lack of loops or conditionals, this vulnerability went unnoticed for over 2 years, leaving Linux machines exposed to remote compromise until its discovery and patching in April 2015. The vulnerability involved unsafe copying of a file name with a length controlled by an attacker, potentially leading to buffer overflow and memory corruption. When code pointers were affected, remote control of the program could be seized by attackers. The paragraph emphasizes the growing need for more robust tools to assist defenders in identifying such subtle yet perilous vulnerabilities within complex codebases in the software industry.

```
phar_set_inode(phar_entry_info *e)
{
  char tmp[MAXPATHLEN];
  int tmp_len;
  tmp_len = e->filename_len + e->phar->fname_len;
  memcpy(tmp, e->phar->fname, e->phar->fname_len);
  memcpy(tmp + e->phar->fname_len, e->filename, e->filename_len);
  e->inode = (unsigned short)zend_get_hash_value(tmp, tmp_len);
}
```

Listing 1.1. An Example of A Vulnerable Function CVE 2015-3329

Table 1 summarizes the four different dimensions of features we studied associated with these vulnerabilities: structure-based, flow-based, binary-based, and pointer-based.

Dimension	Feature	Description
Structure-based	Parameters	number of parameters
	Cyclomatic Complexity	number of linearly independent paths
	Loop Number	number of loops
	Nesting Degree	maximum nesting level of control structures in a function
	SLOC	number of source lines
	Variables	number of local variables
Flow-based	In-degree	number of functions that call the corresponding function
	Out-Degree	number of functions that called by the function
	Height	distance to the closest external data input
Binary based	ALOC	number of assembly codes
	Conditions	number of binary conditions
	Cmps	number of cmp instructions
	Jmps	number of jmp instructions
Pointer based	Pointers	number of pointer variables
	Pointer Arguments	number of pointer arguments
	Pointer Assignments	number of pointer assignments

Table 1. Features

PROPOSED METHOD

Figure 1 illustrates our proposed MCoM framework, containing 4 parts: (1) triplet mixup data augmentation on the minority (vulnerable) class to address imbalance in the tabular security data set; (2) contrastive and feature reconstruction loss to train the encoder and the decoder; (3) pseudo-labeling of the subset of the unlabeled data using a label propagation technique; and (4) downstream tasks that train the predictor (e.g., MLP) with the fixed trained encoder.

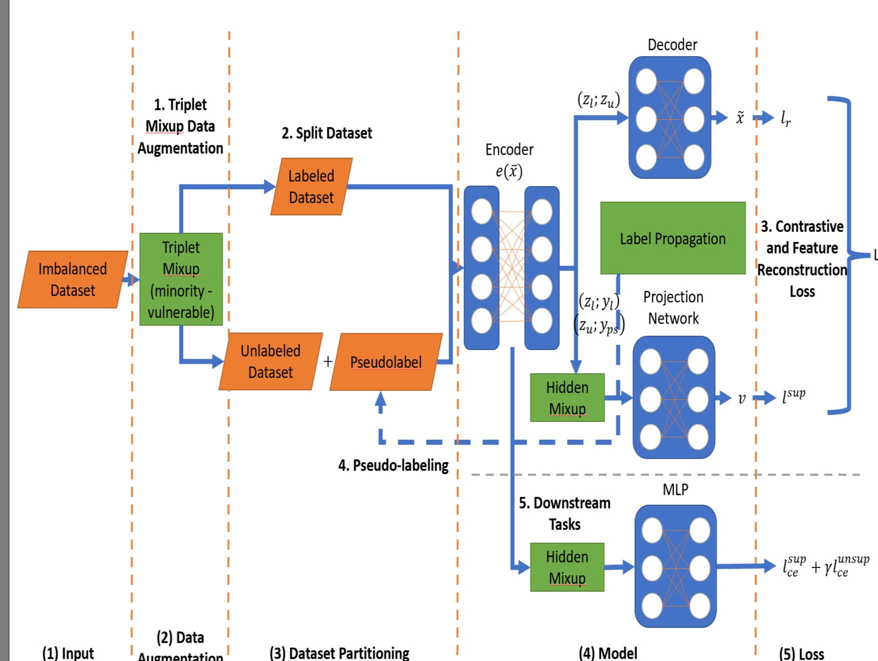

Fig. 1. Overview of MCoM

Mixup trains a neural network on convex combinations of pairs of examples and their labels.

$$\tilde{x} = \lambda x_i + (1-\lambda)x_j, \quad \text{where } x_i, x_j \text{ are raw input vectors}$$
$$\tilde{y} = \lambda y_i + (1-\lambda)y_j, \quad \text{where } y_i, y_j \text{ are one-hot label encodings}$$
$$\lambda \in [0,1]$$

Triplet Mixup Data Augmentation:

$$\hat{x} = \lambda_i x_i + \lambda_j x_j + (1-\lambda_i-\lambda_j)x_k \quad \lambda_i, \lambda_j \sim Uniform(0,\alpha) \text{ with } \alpha \in (0, 0.5]$$

Contrastive and Feature Reconstruction Loss:

$$P(i) = \{p \mid p \in A(i), y_i = \tilde{y}_p\}$$
$$\tilde{h}_{ii}^t = \lambda h_i^t + (1-\lambda)h_i^t \qquad Ne(i) = \{n \mid n \in I, y_i \neq y_n\}$$
$$l_\tau^{sup} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log\left(\frac{\exp(sim(h_i^{proj}, h_p^{proj})/\tau)}{\sum_{n \in Ne(i)} \exp(sim(h_i^{proj}, h_n^{proj})/\tau)}\right) \Rightarrow \text{Contrastive Loss}$$
$$l_r(x_i) = \frac{|C|}{d}\sum_c^{|C|} \|f_\theta(e_\phi(x_i)^c) - x_i^c)\|_2^2 + \frac{|D|}{d}\sum_j^{|D|}\sum_o^{d_{D_j}} \mathbf{1}[x_i^d = o]\log(f_\theta(e_\phi(x_i)^o)) \Rightarrow \text{Reconstruction Loss}$$
$$L = \mathbb{E}_{(x,y)\sim D_L}[l_\tau^{sup}(y, f(x))] + \beta\mathbb{E}_{x \sim D_U \cup D_L}[l_r(x)]$$

Pseudo-labeling:

$$y = \begin{cases} sim(z_i, z_j) & \text{if } i \neq j \text{ and } z_j \in NN_k(i) \\ 0 & \text{otherwise} \end{cases} \quad \tilde{y}_i := \arg\max_j c_{ij} \quad (I - \alpha A)C = Y$$
$$A = D^{-1/2}WD^{-1/2} \qquad W = G^T + G \qquad D := diag(W1_n)$$
$$L = \mathbb{E}_{(x,y)\sim D_L}[l^{sup}(y, f(x))] + \gamma\mathbb{E}_{x, y_{ps}\sim S_U}[l^{sup}(y_{ps}, f(x))] + \beta\mathbb{E}_{x\sim D_U}[l_r(x)]$$

Downstream Tasks:

Cross Entropy Loss $l_{ce}^{sup} + \gamma l_{ce}^{unsup}$

RRESULTS

Table 2 displays experimental outcomes comparing our method with six supervised and two semi-supervised methods (labeled ratio 0.1). We divide the experiments into two sections: (1) without triplet mixup data augmentation (upper part) and (2) with triplet mixup data augmentation (lower part). In the upper section, without triplet mixup data augmentation, all methods predict only negative data and struggle with identifying positive (vulnerable) data, despite high accuracy and TNR. These methods perform poorly on imbalanced tabular security data, regardless of labeled data quantity. In the lower section, employing triplet mixup data augmentation enhances our method, MCoM, achieving the best recall and TPR (66.67%) with just 0.1% labeled data, outperforming other methods, even those using all labeled data. Thus, MCoM exhibits superior performance across all metrics.

						F1 Score		
Model	**Accuracy**	**Precision**	**Recall**	**TPR**	**TNR**	**Micro**	**Macro**	**Weighted**
Without Triplet Mixup Data Augmentation								
Supervised (4554 labeled data)								
XGBoost	**99.21**	0	0	0	**100**	**99.21**	49.80	**98.82**
MLP	**99.21**	0	0	0	**100**	**99.21**	49.80	**98.82**
Logit Regression	**99.21**	0	0	0	**100**	**99.21**	49.80	**98.82**
SVM	**99.21**	0	0	0	**100**	**99.21**	49.80	**98.82**
Decision Tree	98.51	0	0	0	99.29	98.51	49.62	98.46
KNN	99.12	0	0	0	99.91	99.12	49.78	98.77
Semi-supervised (455 labeled data, 0.1 labeled ratio)								
VIME	**99.21**	0	0	0	**100**	**99.21**	49.80	**98.82**
Contrastive Mixup	**99.21**	0	0	0	**100**	**99.21**	49.80	**98.82**
With Triplet Mixup Data Augmentation								
Supervised (17798 labeled data)								
XGBoost	98.68	0	0	0	99.47	98.68	49.67	98.55
MLP	97.54	**4.76**	11.11	11.11	98.23	97.54	**52.71**	98.03
Logit Regression	81.02	2.74	**66.67**	**66.67**	81.13	81.02	47.36	88.79
SVM	89.37	4.10	55.56	55.56	89.64	89.37	51.00	93.67
Decision Tree	97.28	4.17	11.11	11.11	97.96	97.28	52.34	97.89
KNN	91.39	4.12	44.44	44.44	91.76	91.39	51.52	94.79
Semi-supervised (1779 labeled data and 0.1 labeled ratio)								
VIME	78.30	2.40	**66.67**	**66.67**	78.39	78.30	46.19	87.10
MCoM	86.91	3.95	**66.67**	**66.67**	87.07	86.91	50.20	92.28

Table 2. Main experimental results. Top two are shaded and best is bold.

						F1 Score		
Method	**Accuracy**	**Precision**	**Recall**	**TPR**	**TNR**	**Micro**	**Macro**	**Weighted**
With Pairwise Mixup Data Augmentation								
Focal Loss	93.23	5.26	44.44	44.44	93.62	93.23	52.95	95.80
CB Loss	93.06	5.13	44.44	44.44	93.45	93.06	52.79	95.70
Weighted CE	**93.59**	**5.56**	44.44	44.44	**93.98**	**93.59**	**53.28**	**95.99**
With Triplet Mixup Data Augmentation								
MCoM	86.91	3.95	**66.67**	**66.67**	87.07	86.91	50.20	92.28

Table 3. Experimental results with different loss functions.

						F1 Score		
Method	**Accuracy**	**Precision**	**Recall**	**TPR**	**TNR**	**Micro**	**Macro**	**Weighted**
Supervised SVM								
Down Sampling	**92.53**	3.66	33.33	33.33	**93.00**	**92.53**	**51.35**	**95.40**
SMOTE	88.49	3.79	55.56	55.56	88.75	88.49	50.48	93.18
Semi-supervised (0.1 labeled ratio)								
MCoM	86.91	**3.95**	**66.67**	**66.67**	87.07	86.91	50.20	92.28

Table 4. Experimental results with different sampling methods.

						F1 Score		
Method	**Accuracy**	**Precision**	**Recall**	**TPR**	**TNR**	**Micro**	**Macro**	**Weighted**
Pairwise	**93.59**	**5.56**	44.44	44.44	**93.98**	**93.59**	**53.28**	**95.99**
Quadruplet	84.18	2.23	44.44	44.44	84.50	84.18	47.82	90.69
Pairwise+Original	78.21	2.39	**66.67**	**66.67**	78.30	78.21	46.16	87.04
Pairwise+Triplet	85.68	3.61	**66.67**	**66.67**	85.83	85.68	49.55	91.57
Triplet	86.91	3.95	**66.67**	**66.67**	87.07	86.91	50.20	92.28

Table 5. Experimental results with different mixup strategies.

						F1 Score		
Method	**Accuracy**	**Precision**	**Recall**	**TPR**	**TNR**	**Micro**	**Macro**	**Weighted**
No Mixup	99.03	0	0	0	99.82	99.03	49.76	98.73
No Input Mixup	**99.21**	0	0	0	**100**	**99.21**	49.80	**98.82**
No Hidden Mixup	86.29	3.77	**66.67**	**66.67**	86.45	86.29	49.87	91.92
MCoM	86.91	**3.95**	**66.67**	**66.67**	87.07	86.91	**50.20**	92.28

Table 6. Ablation study. Top two are shaded and best is bold.

Table 4 compares our triplet mixup augmentation method with a down-sampling and an up-sampling method (SMOTE). Compared with down-sampling and SMOTE, our method achieves the best recall, TPR (66.67) , and precision (3.95), demonstrating that our method is better overall. Down-sampling reduces negative samples, losing information from the negative samples. In contrast, up-sampling generates more samples from the positive samples. SMOTE generates positive samples by adding small amounts to positive samples. However, our method leverages more information from positive samples by mixing-up triple data points.

Table 5 compares different mixup strategies: pairwise, quadruplet, pairwise + original (mixing a pair of data points including the output of pairwise mixup), pairwise + triplet (mix a pair of data points followed by triplet mixup, including the output of pairwise mixup).

CONCLUSIONS

The paper introduces MCoM, a novel semi-supervised machine learning approach designed for analyzing highly imbalanced tabular security datasets. MCoM consists of four key components: Triplet Mixup Data Augmentation, Contrastive and Feature Reconstruction Loss, Pseudo-labeling, and Downstream Tasks. When compared to six supervised methods and two state-of-the-art semi-supervised methods in tabular domains, it becomes evident that all methods perform poorly without the triplet mixup data augmentation, resulting in zero precision, recall, and TPR. However, upon incorporating the proposed triplet mixup data augmentation, significant improvements are observed, with MCoM achieving the best recall and TPR at 66.67. Future research should explore the applicability of this technique to different datasets in various domains. Additionally, the authors plan to extend their method to graph datasets, including CFGs, CPGs, and ASTs extracted from open-source applications, to identify software vulnerabilities at both source and binary levels. This approach aims to automate the labeling of graphs associated with each function and component, benefiting developers and experts.

REFERENCES

1. Kihyuk Sohn and David Berthelot and Nicholas Carlini and Zizhao Zhang and Han Zhang and Colin A. Raffel and Ekin Dogus Cubuk and Alexey Kurakin and Chun-Liang Li: “FixMatch: Simplifying Semi-supervised Learning with Consistency and Confidence” Advances in Neural Information Processing Systems (NeurIPS), 2020

2. Darabi, S., Fazeli, S., Pazoki, A., Sankararaman, S., Sarrafzadeh, M.: “Contrastive Mixup: Self-and Semi-supervised Learning for Tabular Domain” arXiv Preprint arXiv:2108.12296, 2021

3. Du, X., Chen, B., Li, Y., Guo, J., Zhou, Y., Liu, Y., Jiang, Y.: "Leopard: Identifying Vulnerable Code for Vulnerability Assessment through Program Metrics" Software Engineering (ICSE), 2019

4. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.A.: "MixMatch : A Holistic Approach to Semi-supervised Learning “Advances in Neural Information Processing Systems (NeurIPS), 2019

AKNOWLEDGEMENTS

The research reported herein was supported in part by NSF awards DMS-1737978, DGE-2039542, OAC-1828467, OAC-1931541, and DGE-1906630, ONR awards N00014-17-1-2995 and N00014-20-1-2738, DARPA FA8750-19-C-0006, Army Research Office Contract No. W911NF2110032 and IBM faculty award (Research).

RESEARCH POSTER PRESENTATION DESIGN © 2011
www.PosterPresentations.com