MalwareDNA: Simultaneous Classification of Malware, Malware Families, and Novel Malware

1st Maksim E. Eren
Advanced Research in Cyber Systems
Los Alamos National Laboratory
Los Alamos, USA
maksim@lanl.gov

2nd Manish Bhattarai
3rd Kim Rasmussen
4th Boian S. Alexandrov
Theoretical Division
Los Alamos National Laboratory
Los Alamos, USA

5th Charles Nicholas
CSEE
University of Maryland, Baltimore County
Maryland, USA

***Abstract*—Malware is one of the most dangerous and costly cyber threats to national security and a crucial factor in modern cyber-space. However, the adoption of machine learning (ML) based solutions against malware threats has been relatively slow. Shortcomings in the existing ML approaches are likely contributing to this problem. The majority of current ML approaches ignore real-world challenges such as the detection of novel malware. In addition, proposed ML approaches are often designed either for malware/benign-ware classification or malware family classification. Here we introduce and showcase preliminary capabilities of a new method that can perform precise identification of novel malware families, while also unifying the capability for malware/benign-ware classification and malware family classification into a single framework.**

***Index Terms*—non-negative matrix factorization, malware, semi-supervised learning, reject-option**

I. Introduction

Approximately half a million new malware are reported daily, which drives the increased utilization of Machine Learning (ML) based automated security systems to combat malware [1]. Several ML solutions have previously been introduced for distinct tasks of malware detection and malware family classification. The objective of malware detection is to identify a given file as benign or malicious. In contrast to malware detection, malware family classification assumes that any given sample is already known to be malicious, and we want to know which family it belongs to [2]. Existing solutions often use separate ML systems, where one system may be used for detecting malware, and another system is then used to classify the detected malware into a given family. A system that can unify these tasks would have operational benefits such as reducing the complexity of maintaining separate systems.

In addition, despite its benefits, the adoption of ML-based solutions against malware threats has been relatively slow due to shortcomings in these systems [2]. The majority of the past two decades of research on malware family classification, has not sufficiently accounted for core evaluation criteria including the ability to identify new malware [2], [3]. New malware samples are created regularly by threat actors, which create new versions of already existing malware with identical functionality [2]. Malware analysts regularly go through large quantities of malware samples to understand whether a new malware specimen belongs to a previously known malware family. Classifying a new malware sample into a family or identifying it as *novel* can reduce the number of files analysts need to examine, and aid in understanding the behavior of the malware; this in turn helps estimating the severity of the threat and developing mitigation strategies [2]. At the same time, semi-supervised learning in the malware classification field has not been widely explored despite its superior generalization to new data as compared to supervised systems [2]. With the ever-growing quantity of malware and their complexities there is an urgent need to improve existing solutions and their operational architectures to drive the increased adaption of ML solutions.

Here we introduce a new semi-supervised method, named *MalwareDNA*, that unifies the capability of malware detection and malware family classification into a single framework, while also addressing the shortcomings of novel malware family identification. In this way, MalwareDNA can classify known malware families and separate them from benign-ware, as well as identify new types of families, *all at the same time*. Our method uses hierarchical non-negative matrix factorization (NMF) with automatic model determination [4], which enables data modeling with high specificity and accuracy, to build an archive of latent signatures (identifiers) of malware and benign-ware. These signatures are then be used for precise real-time downstream detection of malware and classification of malware families. Our method also includes a fast optimization method to perform real-time identification of unseen signatures (or novel malware families) by implementation of the *reject-option* method [5]. To the best of our knowledge, we are the first to introduce a framework that combines malware detection, malware family classification, and novel malware family identification capabilities into a single system.

II. Relevant Work

As part of the semi-supervised scheme, our method leverages clustering and similarity scores for categorization of novel samples. A number of previous works have also used clustering approaches, where the ensemble of clustering algorithms with distinct characteristics has been shown to yield accurate results for malware classification [6], [7]. Likewise, similarity metrics to extract embeddings (distance-based fea-

ture vectors) has also shown to be a successful technique for malware analysis [8]. These methods, however, only focus on malware/benign-ware detection or malware family classification, and do not posses the ability to identify novel families.

Several works did consider benign-ware as a class among the classes of malware families [9], [10]. This allowed these methods to separate benign-ware from malware and also classify malware families simultaneously. At the same time, these prior works attempted to detect rare specimens by grouping multiple families into a single *"others"* class. The most realistic malware family classification work was done by Huang et al. which targeted 100 classes where two of the classes include one for benign samples and *"others"* [10]. While this approach introduced an ability to detect rare specimens by the "others" class, it yields poor generalization to new or never before seen specimens as was also pointed out by Loi et al [9]. Loi et al. reports that their false positives are heavily represented by the families collected within the *"others"* class due to the supervised method's inability to learn the patterns of these families from a small number of specimens. Conversely, our method does not require training with rare specimens, instead it posses the abstaining prediction ability (the *reject-option*). This allows our method to uniquely combine the abilities of malware detection and malware family classification, as well as novel malware family identification.

III. Method

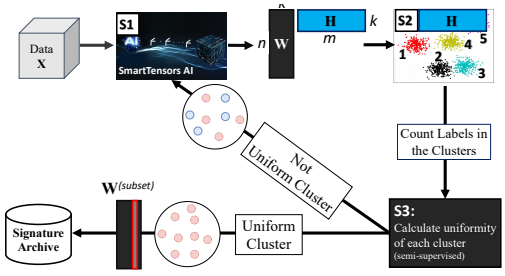

Fig. 1. Building the archive of latent signatures.

A. Building Signature Archive

The overview of how the signature archive is built is shown in Figure 1. MalwareDNA first applies NMF to the observational data $\mathbf{X}$ (**S1**). NMF is an unsupervised learning method based on a low-rank matrix decomposition [11]. NMF approximately represents an observed non-negative matrix, $\mathbf{X} \in \mathbb{R}_{+}^{n \times m}$, as a product of two (unknown) non-negative matrices, $\mathbf{W} \in \mathbb{R}_{+}^{n \times k}$ whose k columns are the latent signatures each with n features, and $\mathbf{H} \in \mathbb{R}_{+}^{k \times m}$ whose rows are the activities of each one of the k signatures (latent features) in each m samples, where usually $k \ll m, n$. This approximation is performed via non-convex minimization constrained by the non-negativity of $\mathbf{W}$ and $\mathbf{H}$: $\min||\mathbf{X}_{ij} - \sum_{s=1}^{k} \mathbf{W}_{is}\mathbf{H}_{sj}||_F$.

The NMF minimization requires prior knowledge of the latent dimensionality k for accurate data modeling, which is usually unavailable [12]. Choosing too small a value of k leads to a poor approximation of the observables in $\mathbf{X}$ (*under-fitting*), while if k is chosen to be too large, the extracted features also fit the noise in the data (*over-fitting*). In this work, we use *NMFk* that incorporates automatic model selection for estimating k [4], [13]. NMFk integrates NMF-minimization with custom clustering and Silhouette statistics, and combines the accuracy of the minimization and robustness/stability of the NMF solutions, using a bootstrap procedure (i.e., generation of a random ensemble of perturbed matrices) is applied to estimate the number of latent features k. MalwareDNA uses a publicly available implementation of NMFk [14].

Next, we apply a custom H-clustering to assign each of the samples (the columns of $\mathbf{X}$) to one of the k signature-clusters (**S2**). In each of these clusters, some of the samples may have different labels (non-uniformity) based on the confidence probability. We evaluate the uniformity of the samples in each cluster, determining whether all labels are the same (**S3**). When a uniform cluster is identified, we separate the samples of this cluster from the data, $\mathbf{X}$, and add the annotated (by the labels) cluster centroid, corresponding column of $\mathbf{W}$, to our archive of signatures. Otherwise, we continue with successive factorizations in a hierarchical manner to separate the mixed latent signatures as shown in Figure 1.

B. Inference Using the Signature Archive

During testing for real-time inference, we project each new sample onto the signature archive using Non-negative Least Squares Solver (NNLS). This allows us to perform real-time identification by representing each new sample as a combination of signatures recorded in the archive and estimating the accuracy, or similarity score, of this representation. We utilize the cosine similarity score of the NNLS projection of the new sample to the signatures in archive. We utilize the similarity scores, together with a threshold, t, to define the malware/benign-ware classification: When a signature possesses a similarity score above t, the labels of the signature will be determined as the classification result. Otherwise, when the similarity score is below t, it will be determined to be a novel malware family ($t = 1.0$ in our experiments).

IV. Experiments

To illustrate the capabilities of MalwareDNA, we randomly sample 1k benign-software and malware specimens from four families (ramnit, adposhel, emotet, and zusy) using a popular benchmark dataset, EMBER-2018 [15]. We select ramnit to represent a malware novel/unseen family. We use the static analysis features byte histogram and entropy, print table distribution, strings entropy, number of strings/exports/imports/sections, file size, and code size.

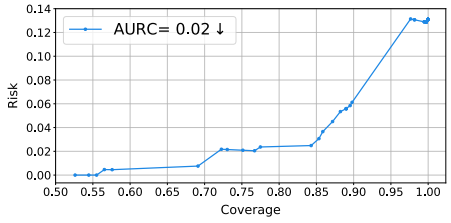

Fig. 2. Risk-Coverage (RC) curve when classifying malware families and the benign-software, together with the area under the RC (AURC).

TABLE I

PERFORMANCE OF MALWAREDNA COMPARED TO BASELINES. REJECTION SEEN PROVIDES THE FALSE REJECTION PREDICTIONS FOR THE SAMPLES THAT BELONGS TO KNOWN CLASSES. REJECTION NOVEL IS THE TRUE REJECTION PREDICTIONS FOR THE SAMPLES THAT BELONGS TO A NOVEL MALWARE FAMILY. XGBOOST+SELFTRAIN AND LIGHTGBM+SELFTRAIN ACHIEVE AURC SCORE OF 0.654 AND 0.651.

Model	F1	Precision	Recall	Rejection Seen	Rejection Novel
MalwareDNA (ours)	**0.975**	**0.975**	**0.977**	15.70 %	**100.00 %**
XGBoost	0.416	0.699	0.510	NA	NA
LightGBM	0.297	0.749	0.338	NA	NA
XGBoost+SelfTrain	0.096	0.258	0.108	4.34 %	18.09 %
LightGBM+SelfTrain	0.096	0.078	0.197	**2.89 %**	17.14 %

The performance of our method is reported with the Area Under the Curve of Risk-Coverage (AURC) [5] in Figure 2. AURC models the trade-off between the coverage (the number of samples for which the non-rejecting predictions were made) and the risk which is measured with 0/1-loss. AURC score is reported between 0 and 1, and lower AURC is preferred over higher AURC. MalwareDNA achieve AURC of 0.02 when classifying the three malware families and benign samples. Our score indicates that we can achieve high coverage with minimal increase in the risk (false rejection predictions).

At 84.3% coverege, MalwareDNA achieves an F1 score of 0.975 when classifying the malware families and the benign-software and 100% true-rejection predictions for the chosen unseen family ramnit, which illustrates our method's ability to identify novel malware families (Table I). In Table 1, we also baseline our method against the state-of-the-art supervised malware classifiers XGBoost [16] and LightGBM [17]. We further extend these baselines with the SelfTrain [18] algorithm to create semi-supervised models. We note that the previous work has used these models to report benchmarking against this dataset [15], [19]; however, we expose these models to a more challenging task of classifying malware families, separating them from benign samples, and detecting novel malware families all at the same time. Our baselines are tuned using Optuna [20] over 100 trials with 5-fold stratified shuffle cross-validation. Our benchmarking against the baseline models and the poor performance of these models, points out both the difficulty of the task, and MalwareDNA' unique capability to both accurately detect malware, classify families, while simultaneously detect novel malware families.

V. CONCLUSION

In this paper, we introduced a new semi-supervised method that unifies three capabilities into a single framework: malware detection, malware family classification, and identification of novel malware families. Our preliminary results showcased the precise novel malware detection capability of our system while also outperforming state-of-the-art methods in a more difficult problem of solving all three inference tasks.

ACKNOWLEDGMENT

This manuscript has been assigned LA-UR-23-25618. This research was funded by the LANL LDRD grant 20230753CR and the LANL Institutional Computing Program, supported by the U.S. Department of Energy National Nuclear Security Administration under Contract No. 89233218CNA000001.

REFERENCES

[1] The Independent IT Security Institute, "Malware statistics & trends report: Av-test," May 2023.

[2] E. Raff and C. Nicholas, "A survey of machine learning methods and challenges for windows malware classification," *ArXiv*, vol. abs/2006.09271, 2020.

[3] A. T. Nguyen, E. Raff, C. Nicholas, and J. Holt, "Leveraging uncertainty for improved static malware detection under extreme false positive constraints," *arXiv preprint arXiv:2108.04081*, 2021.

[4] B. Alexandrov, V. Vesselinov, and K. O. Rasmussen, "Smarttensors unsupervised ai platform for big-data analytics," Los Alamos National Lab.(LANL), Los Alamos, NM (United States), Tech. Rep., 2021, lA-UR-21-25064.

[5] Y. Ding, J. Liu, J. Xiong, and Y. Shi, "Revisiting the evaluation of uncertainty estimation and its application to explore model complexity-uncertainty trade-off," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 4–5.

[6] Y. Ye, T. Li, Y. Chen, and Q. Jiang, "Automatic malware categorization using cluster ensemble," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '10. New York, NY, USA: Association for Computing Machinery, 2010, pp. 95–104.

[7] Y. Zhang, C. Rong, Q. Huang, Y. Wu, Z. Yang, and J. Jiang, "Based on multi-features and clustering ensemble method for automatic malware categorization," in *2017 IEEE Trustcom/BigDataSE/ICESS*, 2017, pp. 73–82.

[8] D. Kong and G. Yan, "Discriminant malware distance learning on structural information for automated malware classification," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '13. New York, NY, USA: Association for Computing Machinery, 2013, pp. 1357–1365.

[9] N. Loi, C. Borile, and D. Ucci, "Towards an automated pipeline for detecting and classifying malware through machine learning," *arXiv preprint arXiv:2106.05625*, 2021.

[10] W. Huang and J. Stokes, "Mtnet: A multi-task neural network for dynamic malware classification," in *Proceedings of 13th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA 2016)*. Springer, July 2016, pp. 399–418.

[11] J.-P. Brunet, P. Tamayo, T. R. Golub, and J. P. Mesirov, "Metagenes and molecular pattern discovery using matrix factorization," *Proceedings of the national academy of sciences*, vol. 101, no. 12, pp. 4164–4169, 2004.

[12] V. Y. Tan and C. Févotte, "Automatic relevance determination in nonnegative matrix factorization with the/spl beta/-divergence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 7, pp. 1592–1605, 2012.

[13] B. T. Nebgen, R. Vangara, M. A. Hombrados-Herrera, S. Kuksova, and B. S. Alexandrov, "A neural network for determination of latent dimensionality in non-negative matrix factorization," *Machine Learning: Science and Technology*, vol. 2, no. 2, p. 025012, 2021.

[14] M. Bhattarai, B. Nebgen, E. Skau, M. Eren, G. Chennupati, R. Vangara, H. Djidjev, J. Patchett, J. Ahrens, and B. ALexandrov, "pydnmfk: Python distributed non negative matrix factorization," 2021.

[15] H. Anderson and P. Roth, "Ember: An open dataset for training static pe malware machine learning models," *ArXiv*, vol. abs/1804.04637, 2018.

[16] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, T. Zhou *et al.*, "Xgboost: extreme gradient boosting," *R package version 0.4-2*, vol. 1, no. 4, pp. 1–4, 2015.

[17] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, 2017.

[18] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proceedings of the 33rd Annual Meeting on Association for Computational Linguistics*, ser. ACL '95. USA: Association for Computational Linguistics, 1995, p. 189–196.

[19] B. Marais, T. Quertier, and C. Chesneau, "Malware analysis with artificial intelligence and a particular attention on results interpretability," in *Distributed Computing and Artificial Intelligence, Volume 1: 18th International Conference 18*. Springer, 2022, pp. 43–55.

[20] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama, "Optuna: A next-generation hyperparameter optimization framework," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 2623–2631.