Microsoft Malware Classification Challenge

Royi Ronen[1], Marian Radu*[2], Corina Feuerstein[1], Elad Yom-Tov[3], and Mansour Ahmadi[4]

[1]Microsoft
[2]CrowdStrike
[3]Microsoft Research
[4]Northeastern University

{royir,corinaf,eladyt}@microsoft.com, marian.radu@crowdstrike.com, m.ahmadi@northeastern.edu

Abstract

The Microsoft Malware Classification Challenge was announced in 2015 along with a publication of a huge dataset of nearly 0.5 terabytes, consisting of disassembly and bytecode of more than 20K malware samples. Apart from serving in the Kaggle competition, the dataset has become a standard benchmark for research on modeling malware behaviour. To date, the dataset has been cited in more than 50 research papers. Here we provide a high-level comparison of the publications citing the dataset. The comparison simplifies finding potential research directions in this field and future performance evaluation of the dataset.

1 Introduction

In recent years, the malware industry has become a large and well-organized market [45]. Well funded, multi-player syndicates heavily invest in technologies and capabilities built to evade traditional protection, requiring anti-malware vendors to develop counter-mechanisms for finding and deactivating them. In the meantime, they inflict significant financial loss to users of computer systems.

One of the major challenges that anti-malware software faces today are the vast amounts of data which needs to be evaluated for potential malicious intent. For example, Microsoft's real-time anti-malware detection products executes on over 600M computers worldwide [36]. This generates tens of millions of daily data points to be analyzed as potential malware. One of the main reasons for these high volumes of different files is that in order to evade detection, malware authors introduce polymorphism to the malicious components. This means that malicious files belonging to the same malware "family", with the same forms of malicious behavior, are constantly modified and/or obfuscated using various tactics, so that they appear to be many different files.

A first step in effectively analyzing and classifying such a large number of files is to group them and identify their respective families. In addition, such grouping criteria may be applied to new files encountered on computers in order to detect them as malicious and of a certain family. To facilitate research in this area, especially in the development of effective techniques for grouping variants of malware files into their respective families, Microsoft provided the data science and security communities with a malware dataset of unprecedented size. Here we summarize the many uses of this dataset, published to date.

2 Dataset

The malware dataset is almost half a terabyte when uncompressed. It consists of a set of known malware files representing a mix of 9 different families. Each malware file has an identifier, a 20 character hash value uniquely identifying the file, and a class label, which is an integer representing one of the 9 family names to which the malware may belong (See Table 1). For each file, the raw

*This work was done while this author was with Microsoft.

Table 1: Malware families in the dataset

Family Name	# Train Samples	Type
Ramnit	1541	Worm
Lollipop	2478	Adware
Kelihos_ver3	2942	Backdoor
Vundo	475	Trojan
Simda	42	Backdoor
Tracur	751	TrojanDownloader
Kelihos_ver1	398	Backdoor
Obfuscator.ACY	1228	Any kind of obfuscated malware
Gatak	1013	Backdoor

data contains the hexadecimal representation of the file's binary content, without the header (to ensure sterility). The dataset also includes a metadata manifest, which is a log containing various metadata information extracted from the binary, such as function calls, strings, etc. This was generated using the IDA disassembler tool. The original question posed to participants was to classify malware to one of the 9 classes. The dataset can be downloaded from the competition website.[1]

3 Citations Comparison

Since the end of the competition in April 2015, more than 50 research papers and thesis works cited the competition and the dataset. Among the citations, several papers are not in English, which we are unable to read [9, 33, 6, 35]. The remaining articles can be divided into two principal classes. The first category of papers referenced the challenge to either perform an abstract comparison or highlight the importance of machine learning for malware classification in industry, where the size of data is huge [43, 19, 28, 47, 18, 38, 49, 44, 25, 53, 46, 21, 4, 57, 16, 17, 39, 50]. Papers in the second category performed partial or complete evaluation on the dataset to verify the effectiveness and/or efficiency of their proposed approach for various tasks. We list the papers of the second category in Table 2 sorted by the publication date. Moreover, we summarize the main contribution or focus of each paper to make higher level clusters. Feature engineering, feature selection/fusion, being scalable, being robust, malware authorship attribution, detecting concept drift, performing a measurement, similarity hashing, classification techniques and deep learning are the major contributions of the papers. The diversity of the contributions has made the dataset a benchmark for various tasks, helping researchers provide a standard for evaluation and comparison.

4 Conclusion and Future Directions

In this paper, we provide a short description of the characteristics of the Microsoft Malware Classification Challenge dataset. This dataset is becoming a standard dataset with more than 50 papers citing it. We enumerated these references as much as possible and compared their main contributions with respect to the dataset. The comparison helps the understanding of what the existing contributions are, and what the potential research directions can be.

The authors aim to keep the reference table updated. We encourage the community to cite this paper when using the dataset, and update us about such work so it can be added to this paper.

References

[1] Mansour Ahmadi, Dmitry Ulyanov, Stanislav Semenov, Mikhail Trofimov, and Giorgio Giacinto. Novel feature extraction, selection and fusion for effective malware family classification. In *Proceedings of the Sixth ACM Conference on Data and Application Security and Privacy*, CODASPY '16, pages 183–194, New York, NY, USA, 2016. ACM. URL: http://doi.acm.org/10.1145/2857705.2857713, doi:10.1145/2857705.2857713.

[1] https://www.kaggle.com/c/malware-classification/data

Month	Year	Method	Focus/Contribution of Research
Mar	2016	Ahmadi et al. [1]	Feature Engineering,Feature Fusion,Being Scalable
May	2016	Drew et al. [13]	Feature Engineering,Being Scalable
Jul	2016	Hu et al. [26]	Being Scalable
Jul	2016	Narayanan et al. [37]	Feature Engineering
Jul	2016	Celik et al. [11]	Being Robust
Aug	2016	Zhang et al. [55]	Being Scalable,Classification Techniques
Sep	2016	Bhattacharya et al. [5]	Being Scalable,Classification Techniques
Oct	2016	Dinh et al. [12]	Classification Techniques
Oct	2016	Wojnowicz et al. [48]	Feature Reduction
Nov	2016	Borbely [7]	Clustering Techniques
Dec	2016	Burnaev et al. [8]	Classification Techniques
Dec	2016	Alrabaee et al. [2]	Malware Authorship Attribution
Jan	2017	Drew et al. [14]	Being Scalable,Classification Techniques
Jan	2017	Patri et al. [40]	Classification Techniques
Mar	2017	Hassen et al. [24]	Feature Engineering,Being Scalable
Mar	2017	Celik et al. [10]	Being Robust
May	2017	Yousefi-Azar et al. [52]	Feature Engineering
Jun	2017	Kebede et al. [30]	Deep Learning
Jul	2017	Yuxin et al. [54]	Deep Learning
Aug	2017	Zhang et al. [56]	Clustering Techniques
Aug	2017	Jordaney at al. [29]	Detecting Concept Drift
Aug	2017	Raff et al. [41]	Similarity Hashing
Oct	2017	Kim et al. [32]	Deep Learning
Nov	2017	Rahul et al. [42]	Deep Learning
Dec	2017	Bagga [3]	Measurement and Comparison
Dec	2017	Gsponer et al. [20]	Classification Techniques
Dec	2017	Hassen et al. [22]	Feature Engineering
Dec	2017	Fan et al. [15]	Being Scalable
Dec	2017	Kim [31]	Deep Learning
Jan	2018	Hwang et al. [27]	Feature Selection
Feb	2018	Yan et al. [51]	Deep Learning
Feb	2018	Kreuk et al. [34]	Adverserial Examples,Deep Learning
Feb	2018	Hassen et al. [23]	Open Set Recognition

Table 2: A comparison between research papers that have performed partial or complete evaluation on Microsoft malware classification challenge dataset.

[2] Saed Alrabaee, Paria Shirani, Mourad Debbabi, and Lingyu Wang. On the feasibility of malware authorship attribution. In Frédéric Cuppens, Lingyu Wang, Nora Cuppens-Boulahia, Nadia Tawbi, and Joaquin Garcia-Alfaro, editors, *Foundations and Practice of Security*, pages 256–272, Cham, 2016. Springer International Publishing.

[3] Naman Bagga. Measuring the effectiveness of generic malware models, 2017.

[4] Thomas Barabosch, Niklas Bergmann, Adrian Dombeck, and Elmar Padilla. Quincy: Detecting host-based code injection attacks in memory dumps. In Michalis Polychronakis and Michael Meier, editors, *Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 209–229, Cham, 2017. Springer International Publishing.

[5] Sukriti Bhattacharya, Héctor D. Menéndez, Earl T. Barr, and David Clark. Itect: Scalable information theoretic similarity for malware detection. *CoRR*, abs/1609.02404, 2016. URL: http://arxiv.org/abs/1609.02404, arXiv:1609.02404.

[6] Philippe Biondi, Xavier Mehrenberger, and Sarah Zennou. Rebus : un bus de communication facilitant la coopération entre outils d'analyse de sécurité. In *Symposium on Information and Communications Security*, 2015.

[7] Rebecca Schuller Borbely. On normalized compression distance and large malware. *Journal of Computer Virology and Hacking Techniques*, 12(4):235–242, Nov 2016. URL: https://doi.org/10.1007/s11416-015-0260-0, doi:10.1007/s11416-015-0260-0.

[8] E. Burnaev and D. Smolyakov. One-class svm with privileged information and its application to malware detection. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pages 273–280, Dec 2016. doi:10.1109/ICDMW.2016.0046.

[9] Evgeny Burnayev and Dmitry Smolyakov. One-class machine of reference vectors using privileged information. In *Information technologies and systems (ITaS)*, 2016.

[10] Z. Berkay Celik, Patrick McDaniel, and Rauf Izmailov. Feature cultivation in privileged information-augmented detection. In *Proceedings of the 3rd ACM on International Workshop on Security And Privacy Analytics*, IWSPA '17, pages 73–80, New York, NY, USA, 2017. ACM. URL: http://doi.acm.org/10.1145/3041008.3041018, doi:10.1145/3041008.3041018.

[11] Z. Berkay Celik, Patrick D. McDaniel, Rauf Izmailov, Nicolas Papernot, and Ananthram Swami. Building better detection with privileged information. *CoRR*, abs/1603.09638, 2016. URL: http://arxiv.org/abs/1603.09638, arXiv:1603.09638.

[12] A. Dinh, D. Brill, Y. Li, and W. He. Malware sequence alignment. In *2016 IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom) (BDCloud-SocialCom-SustainCom)*, pages 613–617, Oct 2016. doi:10.1109/BDCloud-SocialCom-SustainCom.2016.96.

[13] J. Drew, T. Moore, and M. Hahsler. Polymorphic malware detection using sequence classification methods. In *2016 IEEE Security and Privacy Workshops (SPW)*, pages 81–87, May 2016. doi:10.1109/SPW.2016.30.

[14] Jake Drew, Michael Hahsler, and Tyler Moore. Polymorphic malware detection using sequence classification methods and ensembles. *EURASIP Journal on Information Security*, 2017(1):2, Jan 2017. URL: https://doi.org/10.1186/s13635-017-0055-6, doi:10.1186/s13635-017-0055-6.

[15] J. Fan, C. Guan, K. Ren, Y. Cui, and C. Qiao. Spabox: Safeguarding privacy during deep packet inspection at a middlebox. *IEEE/ACM Transactions on Networking*, 25(6):3753–3766, Dec 2017. doi:10.1109/TNET.2017.2753044.

[16] Tonya Fields and Jonathan Graham. Classifying network attack data using random forest. In *CATA*, Dec 2016.

[17] Charles A. Fowler. *A HYBRID INTELLIGENCE/MULTI-AGENT SYSTEM FOR MINING INFORMATION ASSURANCE DATA*. PhD thesis, May 2015.

[18] Ji Gao, Beilun Wang, Zeming Lin, and Weilin Xu Yanjun Qi. Deepcloak: Masking deep neural network models for robustness against adversarial samples. *CoRR*, abs/1702.06763, 2017. URL: http://arxiv.org/abs/1702.06763, arXiv:1702.06763.

[19] Felan Carlo C. Garcia and Felix P. Muga II. Random forest for malware classification. *CoRR*, abs/1609.07770, 2016. URL: http://arxiv.org/abs/1609.07770, arXiv:1609.07770.

[20] Severin Gsponer, Barry Smyth, and Georgiana Ifrim. Efficient sequence regression by learning linear models in all-subsequence space. In Michelangelo Ceci, Jaakko Hollmén, Ljupčo Todorovski, Celine Vens, and Sašo Džeroski, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 37–52, Cham, 2017. Springer International Publishing.

[21] Joachim Hansen. The study of keyword search in open source search engines and digital forensics tools with respect to the needs of cyber crime investigations, 2017.

[22] M. Hassen, M. Carvalho, and P. Chan. Malware classification using static analysis based features. In *IEEE Symposium on Computational Intelligence in Cyber Security*, Dec 2017.

[23] M. Hassen and P. K. Chan. Learning a Neural-network-based Representation for Open Set Recognition. *ArXiv e-prints*, February 2018. arXiv:1802.04365.

[24] Mehadi Hassen and Philip K. Chan. Scalable function call graph-based malware classification. In *Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy*, CODASPY '17, pages 239–248, New York, NY, USA, 2017. ACM. URL: http://doi.acm.org/10.1145/3029806.3029824, doi:10.1145/3029806.3029824.

[25] Tingting He, Jingfeng Xue, Jianwen Fu, Yong Wang, and Chun Shan. Research on malicious code analysis method based on semi-supervised learning. In Ming Xu, Zheng Qin, Fei Yan, and Shaojing Fu, editors, *Trusted Computing and Information Security*, pages 227–241, Singapore, 2017. Springer Singapore.

[26] X. Hu, J. Jang, T. Wang, Z. Ashraf, M. P. Stoecklin, and D. Kirat. Scalable malware classification with multifaceted content features and threat intelligence. *IBM Journal of Research and Development*, 60(4):6:1–6:11, July 2016. doi:10.1147/JRD.2016.2559378.

[27] Seong Oun Hwanga, Trong Kha Nguyenb, and Vu Duc Ly. Feature selection for malware classification. Technical report, January 2018.

[28] Mohammad Imran. *Evlauation of Hidden Markov Model for Malware Behavioural Classification*. PhD thesis, 2016.

[29] Roberto Jordaney, Kumar Sharad, Santanu K. Dash, Zhi Wang, Davide Papini, Ilia Nouretdinov, and Lorenzo Cavallaro. Transcend: Detecting concept drift in malware classification models. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 625–642, Vancouver, BC, 2017. USENIX Association. URL: https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/jordaney.

[30] T. M. Kebede, O. Djaneye-Boundjou, B. N. Narayanan, A. Ralescu, and D. Kapp. Classification of malware programs using autoencoders based deep learning architecture and its application to the microsoft malware classification challenge (big 2015) dataset. In *2017 IEEE National Aerospace and Electronics Conference (NAECON)*, pages 70–75, June 2017. doi:10.1109/NAECON.2017.8268747.

[31] Hae-Jung Kim. Image-based malware classification using convolutional neural network. In James J. Park, Vincenzo Loia, Gangman Yi, and Yunsick Sung, editors, *Advances in Computer Science and Ubiquitous Computing*, pages 1352–1357, Singapore, 2018. Springer Singapore.

[32] Jin-Young Kim, Seok-Jun Bu, and Sung-Bae Cho. Malware detection using deep transferred generative adversarial networks. In Derong Liu, Shengli Xie, Yuanqing Li, Dongbin Zhao, and El-Sayed M. El-Alfy, editors, *Neural Information Processing*, pages 556–564, Cham, 2017. Springer International Publishing.

[33] Nak-Hyun Kim, Byung ik Kim, and Tae jin Lee. Performance analysis of the malware classification method in accordance with the changes in assembly code. 2016.

[34] F. Kreuk, A. Barak, S. Aviv-Reuven, M. Baruch, B. Pinkas, and J. Keshet. Adversarial Examples on Discrete Sequences for Beating Whole-Binary Malware Detection. *ArXiv e-prints*, February 2018. arXiv:1802.04528.

[35] Hyun-Jong Lee, Heo-Jae Hyeok, and Doosung Hwang. Performance Comparison of Machine Learning Algorithms for Malware Detection. volume 26, pages 143–146. The Korean Society Of Computer And Information, 2018. URL: http://www.dbpia.co.kr/Article/NODE07303202.

[36] Microsoft. Sam cybersecurity engagement kit, 2018. URL: https://assets.microsoft.com/en-nz/cybersecurity-sam-engagement-kit.pdf.

[37] B. N. Narayanan, O. Djaneye-Boundjou, and T. M. Kebede. Performance analysis of machine learning and pattern recognition algorithms for malware classification. In *2016 IEEE National Aerospace and Electronics Conference (NAECON) and Ohio Innovation Summit (OIS)*, pages 338–342, July 2016.

[38] A. P. Norton and Y. Qi. Adversarial-playground: A visualization suite showing how adversarial examples fool deep learning. In *2017 IEEE Symposium on Visualization for Cyber Security (VizSec)*, pages 1–4, Oct 2017. doi:10.1109/VIZSEC.2017.8062202.

[39] R. Paranthaman and B. Thuraisingham. Malware collection and analysis. In *2017 IEEE International Conference on Information Reuse and Integration (IRI)*, pages 26–31, Aug 2017. doi:10.1109/IRI.2017.92.

[40] Om Patri, Michael Wojnowicz, and Matt Wolff. Discovering malware with time series shapelets. In *HICSS*, 2017.

[41] Edward Raff and Charles K. Nicholas. Lempel-ziv jaccard distance, an effective alternative to ssdeep and sdhash. *CoRR*, abs/1708.03346, 2017. URL: http://arxiv.org/abs/1708.03346, arXiv:1708.03346.

[42] R. K. Rahul, T. Anjali, Vijay Krishna Menon, and K. P. Soman. Deep learning for network flow analysis and malware classification. In Sabu M. Thampi, Gregorio Martínez Pérez, Carlos Becker Westphall, Jiankun Hu, Chun I. Fan, and Félix Gómez Mármol, editors, *Security in Computing and Communications*, pages 226–235, Singapore, 2017. Springer Singapore.

[43] J. Saxe and K. Berlin. Deep neural network based malware detection using two dimensional binary program features. In *2015 10th International Conference on Malicious and Unwanted Software (MALWARE)*, pages 11–20, Oct 2015. doi:10.1109/MALWARE.2015.7413680.

[44] Daniel Scofield, Craig Miles, and Stephen Kuhn. Fast model learning for the detection of malicious digital documents. In *Proceedings of the 7th Software Security, Protection, and Reverse Engineering / Software Security and Protection Workshop*, SSPREW-7, pages 3:1–3:8, New York, NY, USA, 2017. ACM. URL: http://doi.acm.org/10.1145/3151137.3151142, doi:10.1145/3151137.3151142.

[45] G.S. Shahi, E.F. Pang, and P.P.E. Fong. *Technology in a Changing World*. Lulu Enterprises Incorporated, 2009. URL: https://books.google.co.il/books?id=ZTcIAgAAQBAJ.

[46] Johann Vierthaler, Roman Kruszelnicki, and Julian Schütte. Webeye automated collection of malicious http traffic. Technical report, Fraunhofer Research Institute for Applied and Integrated Security, November 2017.

[47] Beilun Wang, Ji Gao, and Yanjun Qi. A theoretical framework for robustness of (deep) classifiers under adversarial noise. *CoRR*, abs/1612.00334v8, Feb 2017. URL: http://arxiv.org/abs/1612.00334v8, arXiv:1612.00334v8.

[48] M. Wojnowicz, D. Zhang, G. Chisholm, X. Zhao, and M. Wolff. Projecting "better than randomly": How to reduce the dimensionality of very large datasets in a way that outperforms random projections. In *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 184–193, Oct 2016. doi:10.1109/DSAA.2016.26.

[49] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. In *NDSS*, 2018.

[50] Zhiwu Xu, Cheng Wen, Shengchao Qin, and Zhong Ming. Effective malware detection based on behaviour and data features. In Meikang Qiu, editor, *Smart Computing and Communication*, pages 53–66, Cham, 2018. Springer International Publishing.

[51] Jinpei Yan, Yong Qi, and Qifan Rao. Detecting malware with an ensemble method based on deep neural network. *Security and Communication Networks*, Feb 2018.

[52] M. Yousefi-Azar, V. Varadharajan, L. Hamey, and U. Tupakula. Autoencoder-based feature learning for cyber security applications. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 3854–3861, May 2017. doi:10.1109/IJCNN.2017.7966342.

[53] Songqing Yue. Imbalanced malware images classification: a CNN based approach. *CoRR*, abs/1708.08042, 2017. URL: http://arxiv.org/abs/1708.08042, arXiv:1708.08042.

[54] Ding Yuxin and Zhu Siyi. Malware detection based on deep learning algorithm. *Neural Computing and Applications*, Jul 2017. URL: https://doi.org/10.1007/s00521-017-3077-6, doi:10.1007/s00521-017-3077-6.

[55] Y. Zhang, Q. Huang, X. Ma, Z. Yang, and J. Jiang. Using multi-features and ensemble learning method for imbalanced malware classification. In *2016 IEEE Trustcom/BigDataSE/ISPA*, pages 965–973, Aug 2016. doi:10.1109/TrustCom.2016.0163.

[56] 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*, pages 73–82, Aug 2017. doi:10.1109/Trustcom/BigDataSE/ICESS.2017.222.

[57] Morten Oscar Østbye. Multinomial malware classification based on call graphs, May 2017.