ML-generated synthetic data to boost performance of security classifiers

Samuel Fisher
Virginia Tech

Karan Reddy Kandala
Virginia Tech

Harrison Bickerstaff
Virginia Tech

1 Motivation and Problem Statement

Security classifiers are an implementation of machine learning designed to mitigate risks by labeling input data points that show potential malicious intent. Due to higher stakes, these models require careful training in order to reach a higher-than-typical accuracy requirement. When training security classifiers, there are three major data challenges in regards to the training data.

The first data challenge is imbalanced class distributions. The imbalance is caused by a significantly larger amount of benign data points outweighing the smaller amount of malicious data points. This skew is harmful to the classifier during its training process, introducing a bias toward making benign classifications when it is less certain in its classification. In a real world application of a security classifier a bias towards malicious classifications is likely preferable due to the severe risks of a false benign classification.

Another key data challenge concerns limited or underrepresented attack patterns. Malicious activity requiring a security classifier is constantly evolving and not well presented to the public. This means examples of malicious data points are often outdated, filtered, or incomplete. Due to this data challenge security classifier training data sets can often lack robustness, making them susceptible to adversarial manipulation.

The third major data challenge is an insufficient number of training samples. A limited sample size restricts a security classifiers ability to consistently recognize important patterns or resist over fitting. Publicly available security data is scarce due to data collection efforts being hindered by privacy regulations and ethical considerations. This results in significantly smaller data sets for training, validation, and testing compared to other machine learning applications.

We believe that the solution to these challenges is the generation of synthetic samples. Synthetic samples help with the imbalanced class distribution by creating malicious synthetic samples. This increases the population of malicious samples, allowing models trained with additional synthetic data to have a wider range of balances available. The challenge of limited or underrepresented attack patterns is addressed by increasing the number of samples of underrepresented patterns. The increase in the population of underrepresented patterns has the benefit of increasing the classifier robustness, which helps to deal with evolving threats and model drift. Lastly, the insufficient number of training samples is directly addressed by the generation of synthetic samples. These samples can be made publicly available as they are not restricted by the same privacy regulations and ethical concerns.

By using synthetic data generated by TabDDPM to address the presented challenges, we hope to make significant improvements to the reliability, security, and effectiveness of security classifiers.

2 Related Work

Our research will expand upon two previously published papers, "BODMAS: An Open Dataset for Learning Based Temporal Analysis of PE Malware" [4] and "TabDDPM Modelling Tabular Data with Diffusion Models" [1]. These papers explore both the generation of synthetic tabular data and malware classification methods. In the world of malware classification it can be very difficult to keep a highly trained model because of lack of public training data and the intrinsic evolution of malware itself. Copyright concerns for publishing benign software make it hard for researchers to train their classification models on up to date and accurate data.

2.1 BODMAS

BODMAS, an open data set published by a group of researchers at The University of Illinois seeks to help solve some of these problems presented to researchers in the field. The focus of BODMAS was to curate and publish an extensive up to date dataset of benign software and malware to be used in classification training. The researchers were able to put together over 130,000 data points. Not only did the group behind BODMAS look to increase the available sample data, but they also included malware samples from over 580 different families. While other open data sets have a maximum of 9 different families. This well curated dataset provides both the malware binary data and the feature vector data for benign and malicious samples. Due to copyright concerns, the BODMAS team was not able to post the binary data for benign data.

Previous open source datasets have both been too small, and have had zero to little family classification included in the set. The BODMAS team also noticed the lack of updated samples with the majority of other sets only including samples from 2017-2019.

The BODMAS team also used this new data set to assess the impact of "concept drift". The team used old datasets like Ember, SOREL-20M, and UCSB against the BODMAS dataset to measure classifier performance over time. This paper shows that classifiers trained on outdated sets significantly drop in performance when being tested against new sets with an increase of false negatives from new malware families. In the process the BODMAS team shows two strategies to alleviate concept drift. Including, incremental retraining of classifiers using small amounts of new labeled samples, and training new classifiers on updated data.

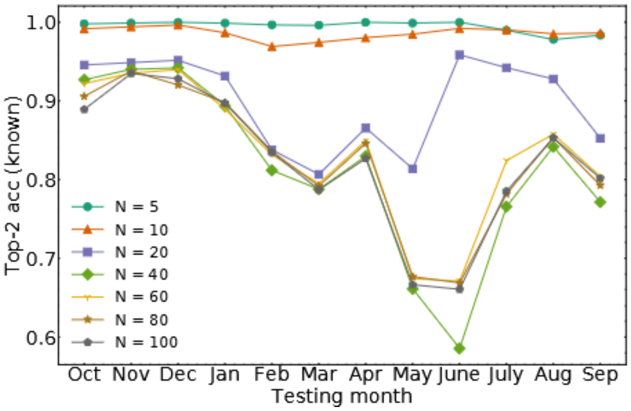

Figure 1: Categorical Accuracy Of Known Families From Bodmas Research

Figure 1, taken from the BODMAS paper, shows the Top-2 acc for classifiers across a simulated year when testing on families that were previously introduced during training. We did note that in the results of BODMAS' work there was a significant drop in performance for training with a family size of N=40. This drop stands out as N=40 is the median value, yet experienced the most extreme drop of tested values. The values represented by this figure are particularly susceptible to underrepresented attack patterns, which likely explains the surprising results of N=40.

Figure 2, taken from the BODMAS paper, shows how classifiers trained on various family sizes perform on test sets that include unseen malware families. This metric also includes previously seen families, representing a testing condition that more accurately reflects the real world. For this figure, the overall performance drops drastically. This significant drop is directly related to the amount of families a classifier was trained on. This again reinforces the importance of having many families in training data, as well as the challenges concept drift represents.

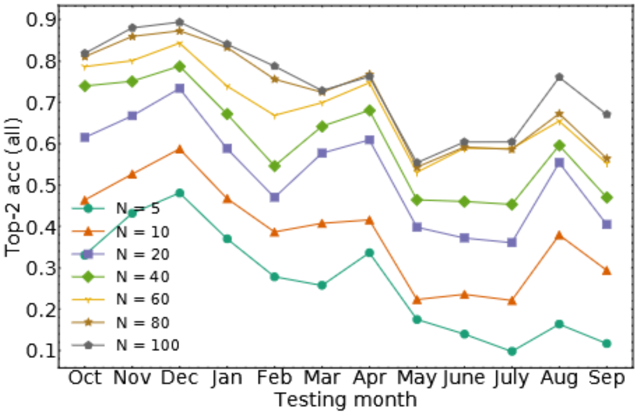

Figure 2: Categorical Accuracy of Known and Unknown Families From Bodmas Research

Although the BODMAS dataset is a leading set researchers use for PE malware Classification there are still limitations that need to be considered. The largest being family or class distributions. Many of the families represented in the BODMAS set are significantly underrepresented. Some families only having a handful of samples as representatives. This can make it a challenge not only to train a classifier to accurately recognize these families but also to train generative models to create additional samples. To magnify these challenges our dataset must be broken into several smaller sets used for training, validation and testing. This further reduces the amount of samples available to train models.

2.2 TabDDPM

We also look to use previous research in artificially generated tabular data to help solve problems when it comes to access of training data. Research conducted by staff at Yandex and the University of Moscow explore new methods for generating synthetic tabular data. This research presents an approach called TabDDPM (tabular data denoising diffusion probabilistic model). This approach is often used to denoise data in fields like computer vision and natural language processing.

The research by TabDDPM uses Gaussian diffusion (Figure 3) for tabular data as well as multi-layer perceptron architecture to reconstruct realistic tabular data. TabDDPM will undergo a process where it adds random noise to data. Once the data becomes unrecognizable TabDDPM will then use diffusion to remove this noise. During this process TabDDPM

$$
\begin{aligned}
&q\left(x_t \mid x_{t-1}\right) := \mathcal{N}\left(x_t; \sqrt{1-\beta_t} x_{t-1}, \beta_t I\right) \\
&q\left(x_T\right) := \mathcal{N}\left(x_T; 0, I\right) \\
&p_\theta\left(x_{t-1} \mid x_t\right) := \mathcal{N}\left(x_{t-1}; \mu_\theta\left(x_t, t\right), \Sigma_\theta\left(x_t, t\right)\right)
\end{aligned}
$$

Figure 3: Gaussian Diffusion

will "learn" how to produce synthetic data starting with random amounts of noise. A good analogy is "learning to paint by restoring old paintings".This data is then restructured into its desired format producing realistic synthetic samples.

Their research suggests that this generated data can be used as a substitute for real data. This is especially useful to us and malware detection models due to the complexity of obtaining accurate training data discussed above. The authors evaluate TabDDPM across 15 tabular datasets, assessing it against other leading generative models like CTGAN, CTABGAN+, and TVAE. The results show that TabDDPM outperforms existing models in capturing feature distributions and correlations. This leads to higher quality generated data, TabDDPM also performs well in machine learning efficiency, meaning classifiers trained on the synthetic data perform well against real-world datasets.

We will use subsets of the BODMAS dataset to tune TabDDPM for more precise generation of synthetic data. Using a series of splits more directly addressed in Section 3, to generate synthetic data that we will use TabDDPM to augment our baseline BODMAS dataset.

2.3 Our Contribution Compared to Related Work

Our research builds directly on previous work with BODMAS and TabDDPM, but diverges by focusing on addressing classifier performance drops specifically associated with underrepresented malware families (n=40 case). While prior studies highlight concept drift and data imbalance, our work introduces an experimental data augmentation method that dynamically increases sample sizes for rare families through synthetic generation.

Additionally, we refine TabDDPM's use by tuning it specifically for malware classification tasks, rather than generic tabular data synthesis. We also evaluate synthetic sample quality not only through classifier metrics but with TabDDPM synthetic quality metrics for additional insight into synthetic malware data quality. This could further lead to the integration of data generation quality control. Targeted augmentation makes our approach novel and directly addresses practical concerns in deploying malware classifiers in real-world, evolving threat environments.

3 Methodology

3.1 Data Extraction Formatting

Our project will generate synthetic data by training TabDDPM on the BODMAS dataset. Before we can begin with any steps, there are a couple of configuration decisions that must be addressed. For example, we can control the number of families that the classifier will use. Based on the data presented in Figure 1 and Figure 2 we have decided to train our classifier with N=40 classes. We choose 40 classes as we believe the severe drop seen in Figure 1 is related to the challenge of underrepresented attack patterns.

Our methodology starts by extracting the BODMAS samples that are used in the 40 families for seed 0. The BODMAS GitHub includes code that will remove the benign samples and any malicious sample not from the 40 families from the dataset. Once the relevant feature vectors have been extracted, we will prepare them for use in training TabDDPM. To do this we separate the data into 3 groups of families: small, medium, and large. We defined small families to be families with a sample quantity of less than 40, medium families to be families with a sample quantity greater than or equal to 40 but less than 100, and large families to be families with a sample quantity greater than or equal to 100.

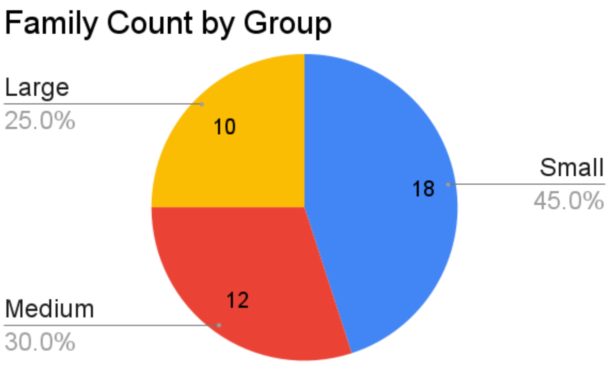

Figure 4: Family Count by Group

Then we create a training, validation, and testing split for each of the family groups and save the features and y labels as separate Numpy files. To create these splits we sampled from each family independently in order to ensure that every family was represented in each split.

3.2 Synthetic Data Generation

Next we tuned TabDDPM to create a configuration file that is optimized specifically for the provided BODMAS dataset. After tuning, TabDDPM is trained and produces the requested amount of synthetic data. The synthetic data is generated in a normalized and unnormalized form, after further testing

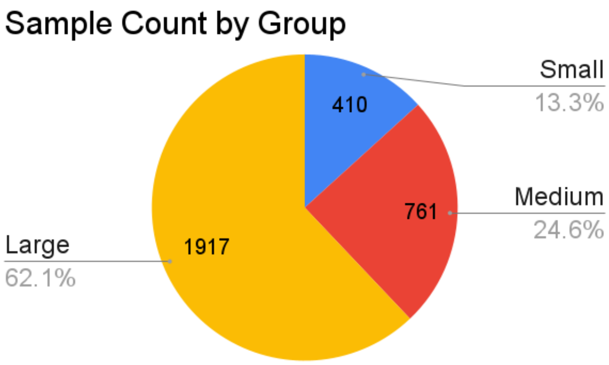

Figure 5: Sample Count by Group

we felt that using the BODMAS data in its original form with the normalized data was the best option. We reached this conclusion by generating lineplot graphs of the features which showed that this combination resulted in both sample types being similar values ranging between 0 and 1 (Figure 6). Whereas other combinations resulted in significantly different ranges between real and synthetic samples.

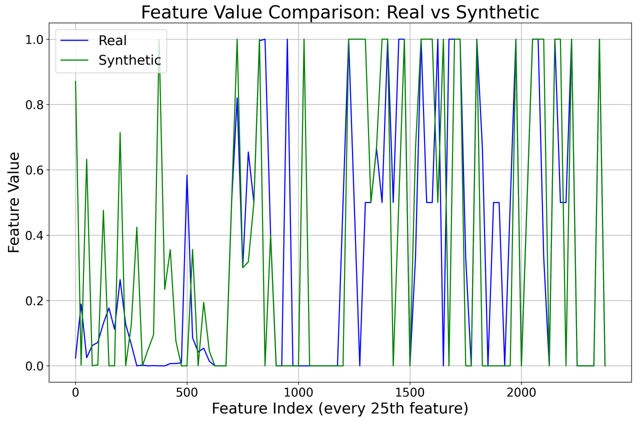

Figure 6: Comparison of the features in a real and synthetic sample

3.3 Dataset Augmentation and Classification

Next, the synthetic data will be merged back together from the three family groups into a singular dataset. Once our complete synthetic dataset is successfully created, we then run the Bluehex classifier. This classifier uses gradient boosted decision trees as a learning method, which provides accuracy but increases the risk of overfitting. This classifier will sample from our synthetic dataset in order to increase the quantity of its training split. The sampling method we implemented allows for two control variables, the max family population and the synth quantity needed. The max family population is calculated for all families, and we found that using a max family population equal to the population of the largest family worked best. The synth quantity needed is calculated individually for each family, with our testing showing that bringing all families to the same population working the best.

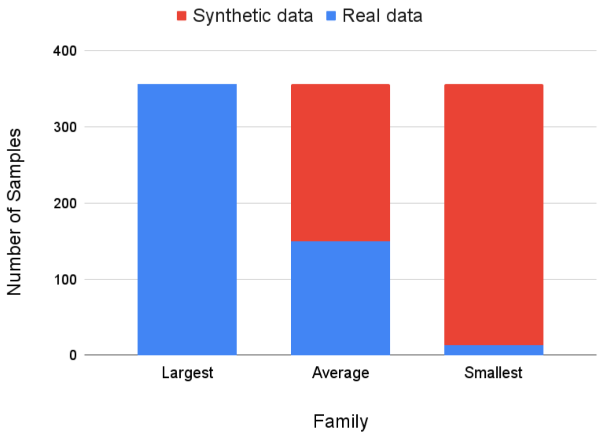

Figure 7: Example of how synthetic data was used to augment the dataset

After we generate our augmented dataset and use it to replace the original dataset, the classifier can be trained, validated, and tested. We also performed another modification to the code of the testing process. This modification is to export the classifiers prediction data, which allows us to calculate evaluation metrics after training and testing is completed. This step in our methodology will be repeated multiple times with different random seeds in order to generate average values and eliminate volatility.

3.4 Evaluation Method

While there is no fixed metric for evaluating real-world performance, we initially intend to use metrics we that would contribute directly to our primary objective. Since our primary objective is to evaluate the performance enhancement that results through the use of generative AI to train and boost the performance of security classifiers. We have decided to establish the BODMAS datasets as our immediate standard. Initial attempt included Recall, Precision, Accuracy, F-1 and Macro F-1 score. We have since centered upon Macro F-1 as the fixed-metric standard for this project. This metric will serve as the ultimate-guide in measuring and evaluating performance. A good macro F1 score is between 0.7-1.0, meaning it can effectively identify positive cases while minimizing false positives and false negatives. F1 score of 0.5-0.7 signifies a struggle to balance the two. Any remaining values fail to effectively classify.

- Macro F-1 Score: Computes F-1 score per class and take their unweighted average, treat all classes equally.

The evaluation criteria is strictly subjective to the increase in Macro(F1) performance. The standard is set by BODMAS and can be observed in Figure 8. Our evaluation objective is to improve performance beyond this baseline, through the use of synthetic TabDDPM datasets. Overall, success will be based primarily on March through May, however, we also seek to improve notable metrics across all months.

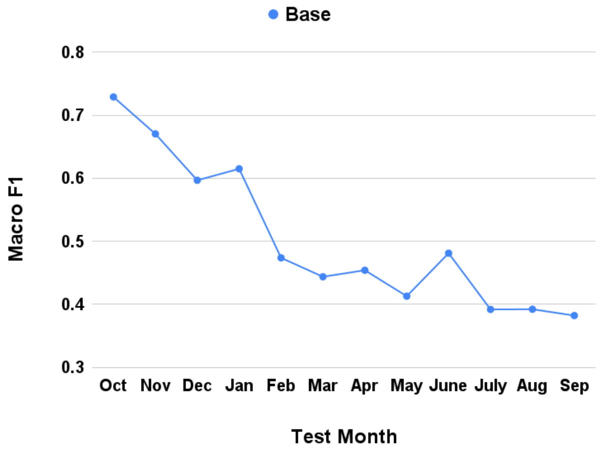

Figure 8: BODMAS Macro-F1 Score Across Testing Months

4 Results

Improvement in the F1 score is the predictor of success. Our evaluated BODMAS data, and augmented data with synthetic samples (tabDDPM) are presented as F1 scores in figure 9. The blue line represents the Macro F1 of the classifier trained on the original BODMAS dataset, while the red line represents the Macro F1 of the classifier trained on our augmented dataset.

In figure 9 Macro F1 scores from the training with the augmented dataset are higher than the base Macro F1 scores, indicating improvement when using synthetic samples. Our results showcase an improved Macro F1 score across the board, an average of 0.056 Macro F1 score improvement which represents a 12.1% increase over the original BODMAS dataset performance. In our key area of interest, March-May, we observe an average increase in Macro F1 of 0.062, representing a 14.6% increase over the original BODMAS dataset performance. We believe this performance increase is not only due to the increase in dataset sample size but more specifically, the equal representation of small malware families in the augmented dataset.

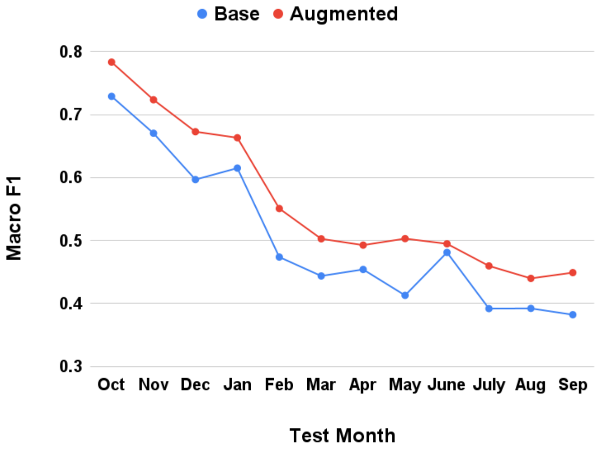

Figure 9: Macro F-1 Comparison between the original BODMAS data and our Augmented Dataset

5 Limitations

Despite improvements, several limitations persist at this stage of our research:

1. **Low amount of BODMAS samples for TabDDPM training:** Due to only using malicious samples from the 40 chosen families, TabDDPM is trained using around 5% of the total malicious samples from BODMAS.

2. **Low Quality Synthetic Samples:** The internal evaluation of the synthetic samples performed by TabDDPM shows consistently low metrics regarding synthetic sample quality, even with extensive tuning measures.

3. **Increase in classifier training time:** The balancing of malicious families significantly increases the training dataset, which has noticeable impact on the training time for the classifier.

6 Future Work

Further work on this project could progress by using these potential improvements:

1. **Improve Synthetic Data Quality:** Explore methods with potential to increase the data quality of the synthetically generated samples.

2. **Post Generation Rejection Sampling:** Use statistical measures to filter poor-quality synthetic samples before training, ensuring only high-fidelity samples augment the training set.

3. **Synthetic Sample Detection:** Evaluate the feasibility of a secondary classifier to differentiate synthetic from

real samples, helping identify potentially low-quality synthetic samples.

4. **Compare with Alternative Synthetic Generation Methods:** Compare results to methods such as SMOTE to evaluate the effectiveness of synthetic in a comparative environment.

7 Statement of Work

Combined contributions:

1. Peer reviewed other sections in the report.
2. Attended help session with Shravya.
3. Attended group meetings on a regular basis.
4. Prepared for presentations in person.
5. Completed submissions on time.
6. Environment creation and organization of group folder on glogin.

Karan Reddy Kandala:

1. Edited and updated Evaluation Plan in the report.
2. Created and presented slides for Demo/Metrics/Evaluation and future goals.
3. Helped with the code issue for evaluation.

Sam Fisher:

1. Wrote the Related Work Limitations and Future Work sections
2. Created and presented slides for the final presentation
3. Backend work for synthetic data quality analysis

Harrison Bickerstaff:

1. Worked on the Motivation/Problem Statement, Methodology, and Results sections in the report.
2. Created and presented slides for Data Generation / Augmentation.
3. Backend work for Demo.
4. Cleaned up and created Project code + readme file.

References

[1] KOTELNIKOV, A., BARANCHUK, D., RUBACHEV, I., AND BABENKO, A. Tabddpm: Modelling tabular data with diffusion models. In *Proceedings of the 40 th International Conference on Machine Learning* (2023), vol. 202.

[2] RAYANKULA, B. C. An evaluation and performance study on bodmas dataset for malware analysis. Masters thesis, National College of Ireland, Dublin, Ireland, 2023. Accessed: 2025-03-07.

[3] ROBINETTE, P. K., LOPEZ, D. M., SERBINOWSKA, S., LEACH, K., AND JOHNSON, T. T. Case study: Neural network malware detection verification for feature and image datasets, 2024.

[4] YANG, L., CIPTADI, A., LAZIUK, I., AHMADZADEH, A., AND WANG, G. Bodmas: An open dataset for learning based temporal analysis of pe malware. In *4th Deep Learning and Security Workshop* (2021).

[1] [2] [3] [4]

We would also like to thank Shravya Kanchi for insightful discussions and valuable feedback on this work.