be influenced by others. We designed an active model and a passive model to compute the weights, and users can choose between the two models by using `--active` or `--passive`. Users can also choose between "01" flips and "10" flips using `--one` or `--zero`.

9) `-vi` or `--visualize-influence`: we use Gephi [1] to visualize the computed influence graph. Similarly, users can choose between the passive model and the active model, and choose between the usage of "01" flips and the usage of "10" flips.

2.3 Querying VTSet

`query.py` is a query tool to select a part of the labeling data based on a user-provided configuration file. `config.html` is a GUI tool to facilitate users to specify their requirements and generate corresponding configuration files. `query.py` can be invoked using a command in the following format:

```
python3 query.py <config> <dataset>
```

<dataset> specifies a sub-directory name under `QResults` to save the selected data. <config> is a json file that describes what labeling data to be selected, and it is generated by `config.html`.

3 RESEARCH OPPORTUNITIES

VTSet captures VirusTotal engines' labeling behavior on a daily basis over a year. Thus, it enables many prediction tasks (by providing training data and ground-truths). We briefly discuss these tasks as follows.

First, can we predict an engine's label on a given file in the near future or in the long run? In our previous work, we identified engines whose labels that are highly correlated with each other and engines whose labeling decisions are highly influenced by other engines [9]. Therefore, it is promising to predict an engine's detection decision on a file based on other engines' detection results on the same file. Since an engine may take some time to react to other engines' behavior, we think it is more reasonable to predict an engine's label on a file after a relatively long period of time.

Second, can we predict the stability of an engine's label on a given file? We found that some engines have many more label flips than other engines, and label flips are more likely to happen on some files [9]. An interesting prediction task is to know how likely an engine would change its label on a file, or whether an engine's label has become stable on the file. If an engine's label on a file has already been stable, the label is more trustworthy, compared with a label that is predicted to be changed in the future.

Third, can we predict whether a file's VirusTotal labels can eventually become stable? In our previous measurements, we found that some files' VirusTotal labels are very difficult to become stable, and VirusTotal engines still change their detection decisions on these files even after the files have been submitted to VirusTotal for more than a year [9]. It is interesting to categorize these files' features and detect files whose VirusTotal labels remain dynamic. For these files, leveraging manual inspection to infer their true labels is more reasonable than using the anti-malware analysis on VirusTotal.

Fourth, can we predict an engine's behavior for a particular malware family? VirusTotal also provides malware family names assigned by its engines. VTSet also records this information. Intuitively, different engines have different capabilities of analyzing different malware families and an engine's behavior on a particular malware family may be different from its overall behavior when analyzing all families. Thus, we can explore how to conduct fine-grained prediction on each malware family using VTSet.

Fifth, how to predict the results of threshold-based label aggregation? Since label changes are very common on VirusTotal, it is important to predict whether a threshold can be met for the number of engines that detect a file. It is promising to leverage our observations in [9] (*e.g.*, highly influenced engines, engines with similar results) to do the prediction.

4 RELATED WORK

There are previous works on evaluating VirusTotal engines or aggregating VirusTotal labels. Peng et al. [7] inspected how VirusTotal's URL scanning engines detect phishing URLs over a month. Kantchelian et al. [5] proposed a machine learning model to aggregate VirusTotal labels. Previous researchers also tried to aggregate malware family names provided by different VirusTotal engines [4, 6, 8]. Different from the data collected by these previous works, VTSet contains daily snapshots of VirusTotal labels on a large set of randomly sampled PE files. VTSet is built through monitoring VirusTotal over one year. Users can observe fine-grained label changes or malware family changes using VTSet and explore how to predict these changes.

5 CONCLUSION

In this paper, we presented VTSet, which is the first available dataset with fine-grained recording of VirusTotal engines' labeling behavior over a long period of time. VTSet can serve as a benchmark set for many prediction tasks on how VirusTotal labels change over time. Future work can consider adding more data to VTSet and extending it to other file types.

ACKNOWLEDGEMENT

This research was supported in part by a Seed Grant award from the Institute for Computational and Data Sciences at the Pennsylvania State University.

REFERENCES

[1] Mathieu Bastian, Sebastien Heymann, and Mathieu Jacomy. Gephi: An open source software for exploring and manipulating networks. In *ICWSM*, 2009.
[2] Kai Chen, Peng Wang, Yeonjoon Lee, XiaoFeng Wang, Nan Zhang, Heqing Huang, Wei Zou, and Peng Liu. Finding unknown malice in 10 seconds: Mass vetting for new threats at the google-play scale. In *USENIX Security*, 2015.
[3] Sean Ford, Marco Cova, Christopher Kruegel, and Giovanni Vigna. Analyzing and detecting malicious flash advertisements. In *ACSAC*, 2009.
[4] Médéric Hurier, Guillermo Suarez-Tangil, Santanu Kumar Dash, Tegawendé F Bissyandé, Yves Le Traon, Jacques Klein, and Lorenzo Cavallaro. Euphony: Harmonious unification of cacophonous anti-virus vendor labels for android malware. In *MSR*, 2017.
[5] Alex Kantchelian, Michael Carl Tschantz, Sadia Afroz, Brad Miller, Vaishaal Shankar, Rekha Bachwani, Anthony D. Joseph, and J. D. Tygar. Better malware ground truth: Techniques for weighting anti-virus vendor labels. In *AISec*, 2015.
[6] Aziz Mohaisen and Omar Alrawi. Av-meter: An evaluation of antivirus scans and labels. In *DIMVA*, 2014.
[7] Peng Peng, Limin Yang, Linhai Song, and Gang Wang. Opening the blackbox of virustotal: Analyzing online phishing scan engines. In *IMC*, 2019.
[8] Marcos Sebastián, Richard Rivera, Platon Kotzias, and Juan Caballero. Avclass: A tool for massive malware labeling. In *RAID*, 2016.
[9] Shuofei Zhu, Jianjun Shi, Limin Yang, Boqin Qin, Ziyi Zhang, Linhai Song, and Gang Wang. Measuring and modeling the label dynamics of online anti-malware engines. In *USENIX Security '20*, Boston, MA, 2020.