Reproduction and Detection of Label-Only Membership Inference Attacks

Xinyu Yang
Department of Computer Science
xinyuyang@vt.edu

1 Introduction

With the development of Artificial Intelligence (AI), machine learning (ML) models have been widely used and provide great convenience to people's lives. However, these ML models may put users' privacy information in danger. Research has proposed that attackers can reveal some privacy information, such as medical records and financial information, by inferring the membership of the train sets of these models. Recently, researchers have proposed label-only membership inference (MI) attacks [Choquette-Choo et al., 2021, Li and Zhang, 2021, Hui et al., 2021]. Compared with score-based MI attacks, these attacks only assume the coarse-grained labels returned by the target model, which pose more threats to real-world ML models.

The fundamental assumption of membership inference attacks is that the training data is more robust than other data. In other words, the members are further away from the decision boundary. Thus, the key to these attacks is to design an efficient method to determine the decision boundary. Existing work [Choquette-Choo et al., 2021, Li and Zhang, 2021] employed query algorithms like HopSkipJump and Sign-OPT [Chen et al., 2020, Cheng et al., 2020], to determine the decision boundary of the sample. If the distance between the sample and the decision boundary exceeds the threshold, then this sample is judged as a member of the training data. Actually, almost all existing label-only membership inference attacks rely on these query algorithms. In other words, the efficiency of membership inference depends on these query algorithms. So, before we evaluate the performance of the membership inference attack, we will first investigate these query algorithms. Furthermore, based on the observation of malicious queries, we proposed a detection approach.

2 Query Algorithms

Various approaches have been proposed to infer the membership information under black-box conditions. The fundamental assumption of these approaches is that the training data is more robust than other data. In other words, the nearest decision boundaries of members are further than those of non-members.

The research proposed by [Brendel et al., 2018] is publicly admitted as the first approach that targets a black-box model. The authors proposed the boundary attack, which attacks the target model based on the random walk. Specifically, the search algorithm starts from an adversarial sample and searches for adversarial examples with random directions. The decision distance is determined by a binary search algorithm for each direction. At the same time, the shortest decision distance is recorded. From Figure 1, we can see that there is a decision boundary for each direction. Finding the shortest decision distance is identical to determining the best direction. After thousands of rounds of search, the algorithm could get a relatively small perturbation value. This value is regarded as the nearest decision boundary for this sample.

Instead of searching the nearest decision boundary randomly, Cheng et al. [2018] proposes a more efficient search algorithm. The authors formulate the boundary searching problem as an optimization problem that can be solved by zeroth-order optimization algorithms (Randomized Gradient-Free

Preprint. Under review.

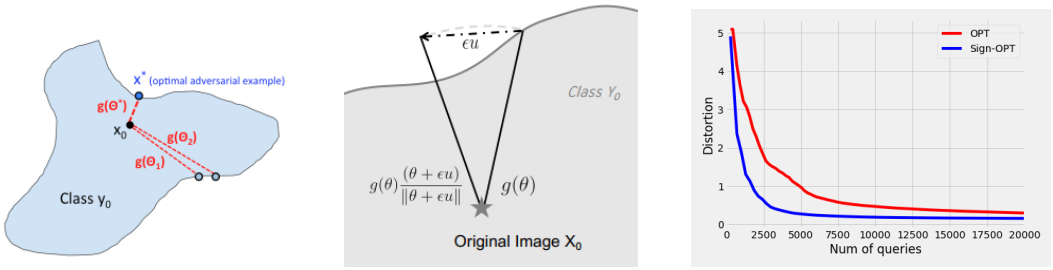

Figure 1: The left figure shows the general principle of boundary distance search algorithms. The middle figure shows the approximation method of Sign-OPT. The right figure is the comparison between them.

Figure 2: The query images of the 1st, 100, 1000, and 5000 rounds

method). Assuming $g(\theta)$ is the decision boundary of search direction θ. To find the direction with the shortest decision distance, the authors approximate the gradient as:

$$\mathbf{g} = \frac{g(\theta + \beta u) - g(\theta)}{\beta} \cdot u \tag{1}$$

In this equation, u is a random vector and β is a smoothing parameter. Each round, the search direction is updated by $\theta_{t+1} = \theta_t - \eta_t \mathbf{g}$.

To further improve the efficiency of this approach, [Cheng et al., 2020] proposed another algorithm. From Equation 1 we can see, to compute the gradient value g, the algorithm has to compute two boundary distance (i.e., $g(\theta + \beta u)$ and $g(\theta)$). However, the authors argued that we do not really need the exact $g(\theta + \beta u)$ value. So they proposed an approximation algorithm, whose principle is shown in the middle of Figure 1. The algorithm computes the sign of another point, which has the same distance from the start point as the original point from the start point. Then the gradient is approximated by the sign of this point, which only needs a single query.

[Chen et al., 2020] goes further on the optimization of search efficiency. This algorithm employs an iterative process including three steps: gradient direction estimation, step-size search, and boundary search. The first two steps are rigorously analyzed, and the third step is the same as in previous research. Based on this algorithm, [Li et al., 2020] proposes three optimization methods. 1) Leveraging spatial transformation to map the data into low-dimensional space; 2) Transferring the image into low-frequency subspace via Discrete Cosine Transformation (DCT); 3) Using PCA to extract the principal components.

3 Defense Algorithm

From the evaluation result, we can see that to launch a membership attack, a large number of queries should be sent to the target model. Moreover, these queries are often similar, which is shown in Figure 2. This feature gives an opportunity to detect these attacks just by inspecting the query stream. Specifically, we can detect this attack based on similarities, like the Euclidean distance and Hamming distance among queries.

However, considering that it is hard to store all the data for a high-speed query stream, a stream processing method should be adopted to detect the attack stream on time. A naive idea is that we can determine whether one query is malicious or not by determining how many similar queries we have received before. However, this approach is hard to conduct for two reasons. First, as we have stated before, we cannot save all queries. Thus, we cannot compute the distance between each pair. That is exactly why we need a stream processing system. Second, we also cannot simply count the number of each query because each malicious query is different from the others even if they are similar.

Given the above challenge, we proposed a KNN detection algorithm with a tumbling window. Specifically, we first initiate a buffering window with size W (e.g., 100) to store queries. Then we collect queries from the query stream and store these queries in the window. When the buffering window is full (i.e., the number of queries equal to the window size W), we evaluate each query in this buffering window. For each query, we find the K nearest neighbors of it in this buffering window. The key insight is that the distance between malicious queries and their nearest neighbors is larger than benign ones. Based on this insight, we compute the average nearest neighbor distance of each query.

4 Evaluation

In this part, we will evaluate the performance of attack and detection algorithms.

4.1 Evaluation of Attack Algorithm

We evaluated the efficiency of these query algorithms. A good query algorithm should determine the shortest decision boundary distance with as few queries as possible. However, different algorithms have different convergence abilities, and different samples have different decision boundary distances. So it is hard to determine a fixed threshold of decision boundary distance. To evaluate the performance of different algorithms. In our evaluation, we fix the query rounds and compare the perturbation distance of different algorithms.

We conducted the evaluation of OPT, Sign-OPT on the CIFAR10 database. For each algorithm, we repeat the query for 1K rounds. The average perturbation distances ($l2$ distance) with different numbers of queries are listed in Table 1.

Table 1: The distortion of OPT and Sign-OPT with different times of queries

	5K Queries	**10K Queries**	**20K Queries**
OPT	0.95	0.46	0.29
Sign-OPT	0.27	0.18	0.16

From this evaluation, we can see that Sign-OPT works better than OPT with the same number of queries. That is because the Sign-OPT uses fewer queries when approximating the gradient. With the same number of queries, Sign-OPT searches more rounds than OPT. Thus, Sign-OPT can get a better result than OPT.

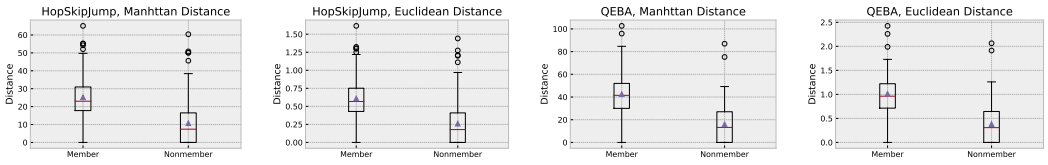

Figure 3: The distance of members and non-members with different searching algorithms and metrics

It is intuitive to achieve the membership inference attacks based on these search algorithms. We just need to measure whether a sample is a training member based on the searched distance. According to the hypothesis, the member samples tend to be more robust than non-member samples, and the more robust a sample is the larger the boundary distance. So we just need to evaluate the distance difference between non-members and members (Figure 3). Then we can set a threshold to differentiate them.

Table 2: The performance of boundary distance-based membership inference attack

Algorithms	Training Datasize	$l0$ distance	$l1$ distance	$l\infty$ distance
HopSkipJump	**3000**	0.8371	0.8358	0.8313
	2000	0.8429	0.8409	0.8364
QEBA	**3000**	0.8792	0.8819	0.8776
	2000	0.8439	0.8426	0.8427

We evaluated the AUC values of different training datasizes and different distance metrics, the result is shown in Table 2. From this table, we can see that the AUC values are more than 80.

4.2 Evaluation of Detection Algorithm

We evaluated the detection algorithm on real attack queries. To extensively evaluate the detection approach, we evaluated the approach with different distance metrics (e.g., Euclidean distance, Hamming distance, and Manhattan distance), different window sizes, and different K values. The evaluation results are shown in Figure 4.

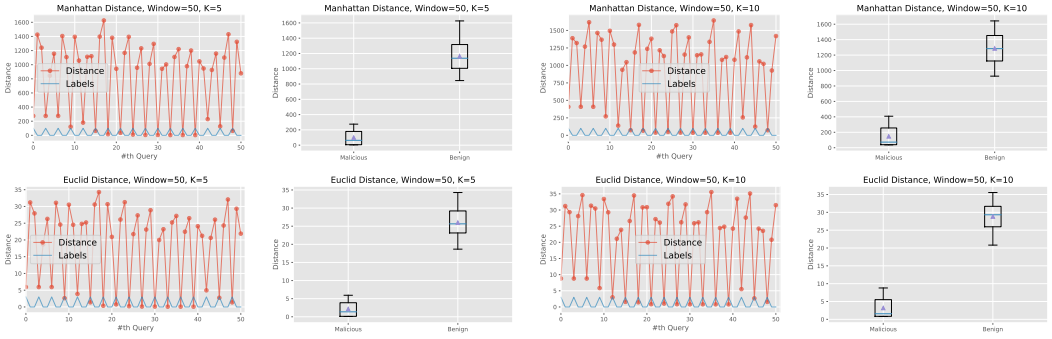

Figure 4: The distances of KNN with different window sizes and K values

The line graphs show the KNN distances and labels of queries. The queries labeled with non-zero values are malicious queries. From the line graphs, we can see that the queries with different labels have apparently different KNN distances. To further evaluate the difference, we plot the box graphs, which make the difference clearer. All in all, we can easily differentiate between malicious and benign queries.

However, there are some limitations to this approach. First, the performance of this approach is relatively low because the computation complexity of a window with W size is $O(W^2)$. Second, the detection approach can only detect malicious queries asynchronously. That is because this approach needs to collect a bunch of queries before it computes the distances. Ideally, a good detection algorithm should be able to detect malicious queries efficiently and synchronously. We will leave it for future work.

5 Conclusion

In this report, we reproduced the membership inference attack and analyzed the principles of such a attack. Then we proposed a detection approach to detect malicious queries based on the similarities between them.

References

Wieland Brendel, Jonas Rauber, and Matthias Bethge. Decision-Based Adversarial Attacks: Reliable Attacks Against Black-Box Machine Learning Models. *arXiv:1712.04248 [cs, stat]*, February 2018.

Jianbo Chen, Michael I. Jordan, and Martin J. Wainwright. HopSkipJumpAttack: A Query-Efficient Decision-Based Attack. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 1277–1294, May 2020. doi: 10.1109/SP40000.2020.00045.

Minhao Cheng, Thong Le, Pin-Yu Chen, Jinfeng Yi, Huan Zhang, and Cho-Jui Hsieh. Query-Efficient Hard-label Black-box Attack:An Optimization-based Approach. *arXiv:1807.04457 [cs, stat]*, July 2018.

Minhao Cheng, Simranjit Singh, Patrick Chen, Pin-Yu Chen, Sijia Liu, and Cho-Jui Hsieh. Sign-OPT: A Query-Efficient Hard-label Adversarial Attack. *arXiv:1909.10773 [cs, stat]*, February 2020.

Christopher A. Choquette-Choo, Florian Tramer, Nicholas Carlini, and Nicolas Papernot. Label-Only Membership Inference Attacks. In *Proceedings of the 38th International Conference on Machine Learning*, pages 1964–1974. PMLR, July 2021.

Bo Hui, Yuchen Yang, Haolin Yuan, Philippe Burlina, Neil Zhenqiang Gong, and Yinzhi Cao. Practical Blind Membership Inference Attack via Differential Comparisons. In *Proceedings 2021 Network and Distributed System Security Symposium*, Virtual, 2021. Internet Society. ISBN 978-1-891562-66-2. doi: 10.14722/ndss.2021.24293.

Huichen Li, Xiaojun Xu, Xiaolu Zhang, Shuang Yang, and Bo Li. QEBA: Query-Efficient Boundary-Based Blackbox Attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1221–1230, 2020.

Zheng Li and Yang Zhang. Membership Leakage in Label-Only Exposures. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 880–895, Virtual Event Republic of Korea, November 2021. ACM. ISBN 978-1-4503-8454-4. doi: 10.1145/3460120.3484575.