

Model Extraction and Membership Inference Attacks on Deployed Models

Juby George

Assistant Professor, Department of Computer Applications, Marian College Kuttikkanam Autonomous, India

Article information

Received: 2nd June 2026

Received in revised form: 4th July 2026

Accepted: 5th August 2026

Available online: 12th September 2026

Volume: 1

Issue: 3

DOI: <https://doi.org/10.5281/zenodo.22702167>

Abstract

A prediction interface gives away more than a prediction. Query it enough times and an attacker can rebuild a working copy of the model, decide whether a particular record was in its training set, or in some settings recover the record itself. This review covers thirty three works and separates two attack families that are often treated as one. Extraction targets the model as intellectual property; inference targets the people whose data trained it. We trace the progression from the first prediction API attack to modern likelihood ratio tests that succeed at low false positive rates, and we explain why average case accuracy is the wrong way to report a membership attack. On the defensive side we group techniques into prevention, obstruction and attribution, and we note that only the first carries a formal guarantee. Two conclusions stand out. Overfitting drives membership leakage but does not fully explain it. And watermarking, the industry favourite for extraction, has weaker guarantees than its adoption would suggest.

Keywords:- Model Extraction, Membership Inference, Machine Learning Privacy, Differential Privacy, Model Watermarking, Prediction API, Threat Model.

I. INTRODUCTION

Machine learning is sold as a service. A client sends an input and receives a prediction, and the weights stay on the provider side. That arrangement is supposed to protect two things at once: the provider's investment in training, and the privacy of whoever supplied the training data. It protects neither as well as it appears to.

Tramer and colleagues showed the first half in 2016. With enough queries to a prediction API you can reconstruct a model that behaves like the original, sometimes exactly, and the query budget was small enough to be affordable [3]. Shokri and colleagues showed the second half a year later: given only the output for a record, an attacker can often tell whether that record was used in training [5]. For a model trained on medical records, that is a disclosure about a person, not about a model.

The two attacks are frequently discussed together and they have different victims. Extraction is a theft of intellectual property. Inference is a privacy breach. The defences differ accordingly, and conflating them leads to the mistake of deploying a watermark and believing the training data are now protected. Figure 1 lays out the shared attack surface. Three surveys cover parts of this ground [26], [29], [31]; our aim is to state the distinction sharply and to be direct about what the defences guarantee.

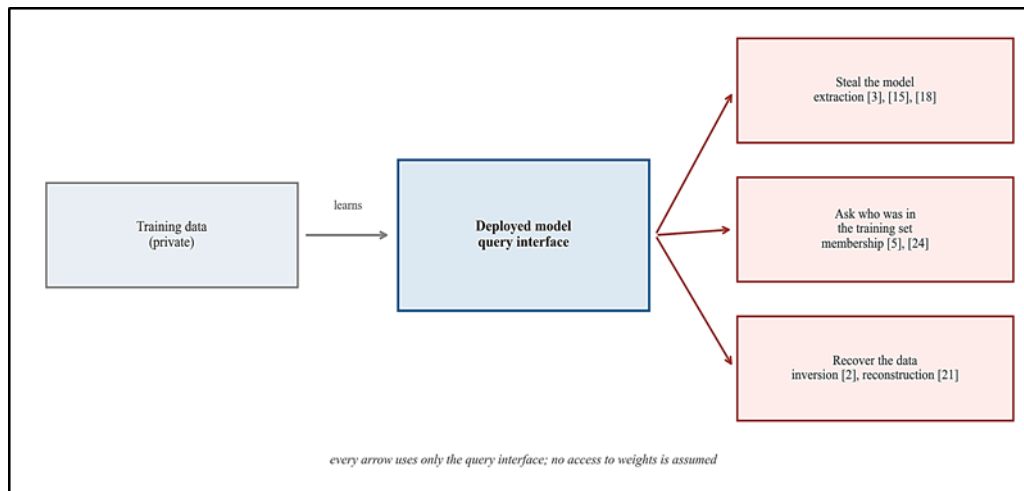


Figure. 1: One interface, three goals. Extraction targets the model, inference targets the training data, and both work through the same query channel.

II. STEALING THE MODEL

Extraction has two goals that pull in different directions. High accuracy extraction wants a model that performs well on the task. High fidelity extraction wants a model that agrees with the target on every input, including the inputs where the target is wrong. Jagielski and colleagues drew that distinction and showed the second is much harder [18].

Method matters less than query strategy. Knockoff nets query the target with images drawn from an unrelated public pool, then train a student on the returned scores, and reach useful accuracy without any data from the original distribution [15]. Chandrasekaran and colleagues made the connection to active learning explicit, which reframes extraction as a query efficiency problem with a well studied theory behind it [19]. For simple model classes, exact recovery through equation solving is possible [3].

Defences fall into three layers, shown in Figure 2. Obstruction perturbs the returned confidence scores, trading a little utility for a worse training signal [17], or watches the distribution of incoming queries for the unnatural patterns that extraction produces [14]. Attribution accepts that the model may be stolen and tries to prove it afterwards, which is what watermarking does: embed a signal in the weights [7], or in the behaviour on a secret trigger set [9], or entangle the mark with the task itself so that removing it costs accuracy [23].

The uncomfortable finding is about watermarking. Lukas and colleagues evaluated image classification watermarks against a broad set of removal attacks and found that most schemes fail under an adversary who is willing to spend a modest amount of compute [25]. Watermarking is worth deploying. It is not worth relying on.

Table 1. Extraction attacks and defences, with what each one assumes or provides, following the cited papers.

Technique	Type	Assumption or guarantee	Source
Equation solving	Attack	Simple model class, score outputs	[3]
Knockoff nets	Attack	Unrelated public query pool	[15]
Active learning query design	Attack	Query budget is the binding cost	[19]
Output perturbation	Defence	Utility loss, no guarantee	[17]
Query monitoring	Defence	Detects abnormal query patterns	[14]
Watermarking	Defence	Attribution only, removal shown feasible	[7], [9], [23], [25]

III. INFERRING THE TRAINING DATA

A. Membership inference

The question is narrow and the consequences are not. Was this record in the training set? Shokri and colleagues answered it by training shadow models on data from a similar distribution, then training a classifier to recognise the output patterns a model produces for members [5]. Salem and colleagues stripped away most of the assumptions and showed the attack still works with a single shadow model, or none [12].

Yeom and colleagues connected leakage to overfitting, which gave the field a useful intuition and, taken too far, a misleading one [8]. Overfitting is sufficient for membership leakage. It is not necessary. Well generalised models still leak on their atypical training points, which are exactly the records that identify individuals.

Choquette-Choo and colleagues removed the last convenient assumption by mounting the attack with only the predicted label, no confidence scores [20]. Nasr and colleagues extended the analysis to white box and federated settings, where an adversary who sees updates has more to work with than one who sees outputs [13]. Zhu and colleagues showed that shared gradients alone can leak the training input [16], which bears directly on federated deployments [33].

B. Reporting these attacks honestly

Carlini and colleagues made an argument that changed how this work should be reported [24]. A membership attack with fifty five percent average accuracy sounds weak. But an attack that is right about a small number of records with near certainty is a serious breach, and average accuracy hides it. They recommend reporting the true positive rate at a low fixed false positive rate, and evaluating per example rather than in aggregate. Song and Mittal reached a similar conclusion from a different direction [22], and Ye and colleagues built a hypothesis testing framework around it [28].

Anyone reading older results should apply the correction. Papers reporting only average accuracy understate the risk for the most vulnerable records, and those are the records that matter.

C. Beyond membership

Model inversion reconstructs a representative input for a class rather than deciding membership, which Fredrikson and colleagues demonstrated on face recognition [1]. Property inference recovers global facts about the training distribution that the provider never intended to publish [10]. Extraction of verbatim training data is now documented for large language models [21] and for diffusion models, where individual training images can be regenerated [30]. Tramer and colleagues added a poisoning step that raises the leakage of targeted records, which turns a passive attacker into an active one [27]. Table 2 sets out the family.

Table 2. Inference attacks grouped by what the adversary learns and what access is required, as reported in the cited works.

Attack	What is learned	Access needed	Source
Membership inference	Was this record trained on	Scores, or labels only	[5], [12], [20]
White box and federated	Membership, from updates	Model internals or gradients	[13], [16]
Model inversion	A class representative input	Scores	[1]
Property inference	Facts about the training set	Model access	[10]
Verbatim extraction	Actual training records	Generation interface	[21], [30]

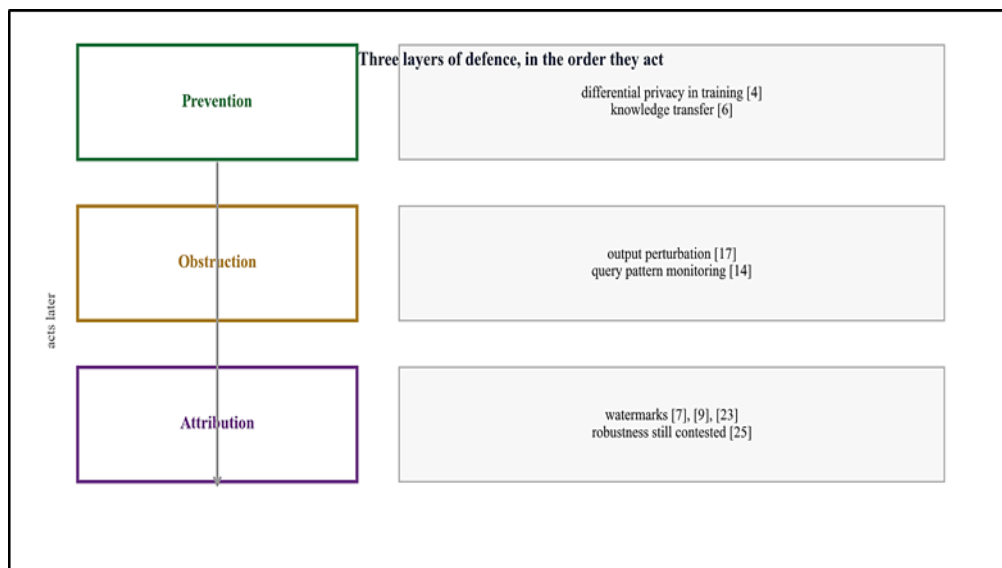


Figure. 2: Defences grouped by when they act. Only the prevention layer carries a formal guarantee.

IV. DEFENCES AND WHAT THEY GUARANTEE

Differential privacy is the only prevention with a formal guarantee. The definition bounds how much any single record can change the output distribution [2], and DP-SGD applies it to deep learning by clipping per example gradients and adding calibrated noise [4]. PATE reaches a similar end through knowledge transfer from an ensemble of teachers trained on disjoint partitions [6]. The guarantee is real and it is bought with accuracy, particularly on the rare classes that matter most in medical and financial work.

Empirical defences carry no such guarantee. Adversarial regularisation trains the model against a membership attacker and reduces leakage against the attacks tested [11]. Whether it holds against a stronger attacker is the same open question that dogs empirical defences elsewhere in security. Encryption based approaches change the picture differently by keeping data confidential during computation rather than bounding what the model memorises [32].

A note on privacy budgets. The epsilon values used in deployed systems are often far larger than the theory's comfortable range, and a large epsilon means the guarantee is weak even though it is technically present. Reporting the value, rather than reporting that differential privacy was used, is the minimum a reader needs.

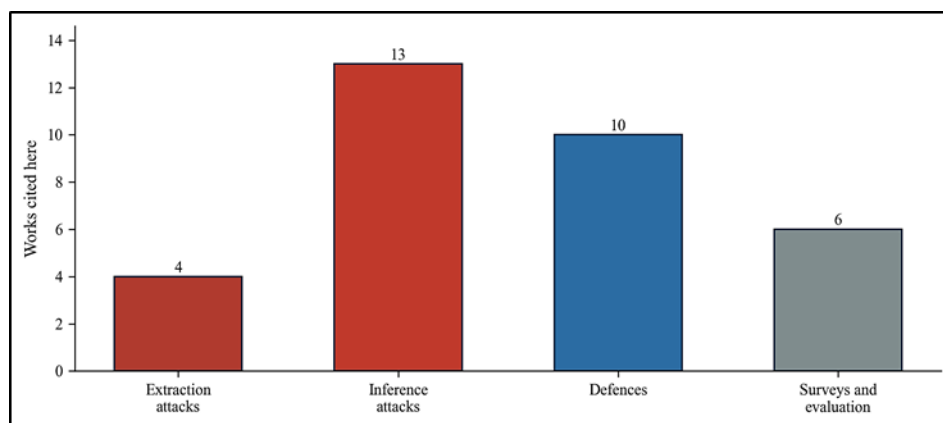


Figure. 3: The thirty three works cited here, coded by the role they play in this review. The chart counts our own reference list and reports no experimental result.

V. OPEN PROBLEMS

Auditing comes first. Given a deployed model, how much did it actually leak? Formal bounds are loose and empirical attacks are lower bounds, so the true value sits in a wide interval. Second, generative models. Extraction and memorisation behave differently there [21], [30], and the defences designed for classifiers do not transfer cleanly.

Third, the cost of privacy falls unevenly, and the classes that lose most accuracy under DP-SGD tend to be the small ones. Fourth, the legal position. A right to erasure implies removing a record's influence from a trained model, and no reviewed method achieves that without retraining.

VI. CONCLUSION

A prediction interface is an information channel, and this literature is the study of what flows through it. Extraction and inference use the same channel and threaten different parties, so a provider needs a position on both.

Three practical points close the review. Use differential privacy where the training data are personal, and report epsilon rather than the fact of its use [2], [4]. Evaluate membership risk at low false positive rates, not on average [24]. And treat watermarking as evidence for a later dispute rather than as a defence, given how it performs under removal attacks [25].

REFERENCES

- [1] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proc. ACM Conf. Comput. Commun. Security (CCS)*, 2015, pp. 1322–1333.
- [2] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proc. Theory of Cryptography Conf. (TCC)*, 2006, pp. 265–284.

- [3] F. Tramer, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Stealing machine learning models via prediction APIs," in *Proc. USENIX Security Symp.*, 2016, pp. 601–618.
- [4] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," in *Proc. CCS*, 2016, pp. 308–318.
- [5] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *Proc. IEEE Symp. Security and Privacy (SP)*, 2017, pp. 3–18.
- [6] N. Papernot, M. Abadi, U. Erlingsson, I. Goodfellow, and K. Talwar, "Semi-supervised knowledge transfer for deep learning from private training data," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2017.
- [7] Y. Uchida, Y. Nagai, S. Sakazawa, and S. Satoh, "Embedding watermarks into deep neural networks," in *Proc. ACM Int. Conf. Multimedia Retrieval (ICMR)*, 2017, pp. 269–277.
- [8] S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha, "Privacy risk in machine learning: Analyzing the connection to overfitting," in *Proc. IEEE Comput. Security Found. Symp. (CSF)*, 2018, pp. 268–282.
- [9] Y. Adi, C. Baum, M. Cisse, B. Pinkas, and J. Keshet, "Turning your weakness into a strength: Watermarking deep neural networks by backdooring," in *Proc. USENIX Security Symp.*, 2018, pp. 1615–1631.
- [10] K. Ganju, Q. Wang, W. Yang, C. A. Gunter, and N. Borisov, "Property inference attacks on fully connected neural networks using permutation invariant representations," in *Proc. CCS*, 2018, pp. 619–633.
- [11] M. Nasr, R. Shokri, and A. Houmansadr, "Machine learning with membership privacy using adversarial regularization," in *Proc. CCS*, 2018, pp. 634–646.
- [12] A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes, "ML-Leaks: Model and data independent membership inference attacks and defenses on machine learning models," in *Proc. Netw. Distrib. Syst. Security Symp. (NDSS)*, 2019.
- [13] M. Nasr, R. Shokri, and A. Houmansadr, "Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning," in *Proc. SP*, 2019, pp. 739–753.
- [14] M. Juuti, S. Szyller, S. Marchal, and N. Asokan, "PRADA: Protecting against DNN model stealing attacks," in *Proc. IEEE Eur. Symp. Security and Privacy (EuroS&P)*, 2019, pp. 512–527.
- [15] T. Orekondy, B. Schiele, and M. Fritz, "Knockoff nets: Stealing functionality of black-box models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 4949–4958.
- [16] L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2019, pp. 14747–14756.
- [17] T. Orekondy, B. Schiele, and M. Fritz, "Prediction poisoning: Towards defenses against DNN model stealing attacks," in *Proc. ICLR*, 2020.
- [18] M. Jagielski, N. Carlini, D. Berthelot, A. Kurakin, and N. Papernot, "High accuracy and high fidelity extraction of neural networks," in *Proc. USENIX Security Symp.*, 2020, pp. 1345–1362.
- [19] V. Chandrasekaran, K. Chaudhuri, I. Giacomelli, S. Jha, and S. Yan, "Exploring connections between active learning and model extraction," in *Proc. USENIX Security Symp.*, 2020, pp. 1309–1326.
- [20] C. A. Choquette-Choo, F. Tramer, N. Carlini, and N. Papernot, "Label-only membership inference attacks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2021, pp. 1964–1974.
- [21] N. Carlini *et al.*, "Extracting training data from large language models," in *Proc. USENIX Security Symp.*, 2021, pp. 2633–2650.
- [22] L. Song and P. Mittal, "Systematic evaluation of privacy risks of machine learning models," in *Proc. USENIX Security Symp.*, 2021, pp. 2615–2632.
- [23] H. Jia, C. A. Choquette-Choo, V. Chandrasekaran, and N. Papernot, "Entangled watermarks as a defense against model extraction," in *Proc. USENIX Security Symp.*, 2021, pp. 1937–1954.
- [24] N. Carlini, S. Chien, M. Nasr, S. Song, A. Terzis, and F. Tramer, "Membership inference attacks from first principles," in *Proc. SP*, 2022, pp. 1897–1914.
- [25] N. Lukas, E. Jiang, X. Li, and F. Kerschbaum, "SoK: How robust is image classification deep neural network watermarking?," in *Proc. SP*, 2022, pp. 787–804.
- [26] H. Hu, Z. Salicic, L. Sun, G. Dobbie, P. S. Yu, and X. Zhang, "Membership inference attacks on machine learning: A survey," *ACM Computing Surveys*, vol. 54, no. 11s, pp. 235:1–235:37, 2022.
- [27] F. Tramer, R. Shokri, A. San Joaquin, H. Le, M. Jagielski, S. Hong, and N. Carlini, "Truth serum: Poisoning machine learning models to reveal their secrets," in *Proc. CCS*, 2022, pp. 2779–2792.
- [28] J. Ye, A. Maddi, S. K. Murakonda, V. Bindschaedler, and R. Shokri, "Enhanced membership inference attacks against machine learning models," in *Proc. CCS*, 2022, pp. 3093–3106.
- [29] D. Oliynyk, R. Mayer, and A. Rauber, "I know what you trained last summer: A survey on stealing machine learning models and defences," *ACM Computing Surveys*, vol. 55, no. 14s, pp. 324:1–324:41, 2023.
- [30] N. Carlini *et al.*, "Extracting training data from diffusion models," in *Proc. USENIX Security Symp.*, 2023, pp. 5253–5270.
- [31] M. Rigaki and S. Garcia, "A survey of privacy attacks in machine learning," *ACM Computing Surveys*, vol. 56, no. 4, pp. 101:1–101:34, 2024.
- [32] M. J. Komban, "Privacy-preserving techniques in data mining: A comprehensive analysis of homomorphic encryption and differential privacy approaches," *Int. J. Inf. Technol. Res. Stud. (IJITRS)*, vol. 1, no. 2, p. 127, Jul. 2025, doi: 10.63090/ijitrs/3139.3209.0011.
- [33] K. Abraham, "Federated learning: Privacy-preserving distributed training," *Eduschool J. Comput. Sci. Res. Stud. (EJCSRS)*, vol. 1, no. 1, pp. 22–26, Mar. 2026, doi: 10.5281/zenodo.18979096.