CSC 592 Project Proposal

Calvin Higgins

February 2025

1 Background

1.1 Approximate Membership Query Filters

Approximate membership query (AMQ) filters on a set S provide a query method Q such that

1. If $x \in S$, $Q(x)$ outputs $x \in S$.
2. If $x \notin S$, $\mathbb{P}[Q(x) \text{ outputs } x \in S] \leq \epsilon$.

where $0 < \epsilon < 1$. That is, the query method Q checks if $x \in S$ with bounded false positive rate ϵ.

1.2 Naor-Eylon Filters

Naor-Eylon filters [3] provably defend AMQ filters against probabilistic polynomial-time (PPT) adversaries. They store the set $F(S)$ in an AMQ filter where F is a pseudo-random permutation (PRP). Let Q_{NE} be the Naor-Eylon filter method and Q_{AMQ} be the AMQ query method. Then

1. For all x, $Q_{\mathrm{NE}}(x)$ outputs $Q_{\mathrm{AMQ}}(F(x))$.

If there exists a PPT adversary that non-negligibly degrades the false positive rate, then PRPs do not exist and $\mathbf{P} = \mathbf{NP}$.

1.3 Learned Bloom Filters

Learned Bloom filters (LBFs) [2] augment standard AMQ filters with a machine learning model Q_{M} that filters "easy" queries. Given an element x, Q_{M} is trained to output $x \in S$. As Q_{M} might have non-zero false negative rate, negative queries are checked against an AMQ filter containing all false negatives. Let Q_{LBF} be the LBF query method and Q_{AMQ} be the AMQ filter query method. Then

1. If $Q_{\mathrm{M}}(x)$ outputs $x \in S$, $Q_{\mathrm{LBF}}(x)$ outputs $x \in S$.
2. If $Q_{\mathrm{M}}(x)$ outputs $x \notin S$, $Q_{\mathrm{LBF}}(x)$ outputs $Q_{\mathrm{AMQ}}(x)$.

where the Q_{AMQ} false positive rate ϵ_{AMQ} is derived from the desired false positive rate ϵ_{LBF}. In short, LBFs first predict if $x \in S$ with a machine learning model M and then recover the bounded single-tailed error by querying a standard AMQ filter.

2 Summary

The paper [1] introduces a secure LBF variant called the Downtown Bodega filter (DBF). The DBF verifies the machine learning model Q_{M}'s output with two Naor-Eylon filters. One filter contains all true positives while the other contains all false negatives. Let Q_{DBF} be the DBF query method, Q_{TP} be the true positive filter, and Q_{FN} be the false negative filter. Then

1. If $Q_{\mathrm{M}}(x)$ outputs $x \in S$, $Q_{\mathrm{DBF}}(x)$ outputs $Q_{\mathrm{TP}}(x)$.
2. If $Q_{\mathrm{M}}(x)$ outputs $x \notin S$, $Q_{\mathrm{DBF}}(x)$ outputs $Q_{\mathrm{FN}}(x)$.

The authors prove that DBFs are secure against PPT adversaries, and simulate the expected false positive rate of DBFs and Naor-Eylon filters on real datasets.

3 Threat Model

Under the proposed threat model, the attacker provides a PPT adversary which

1. Constructs the set S.
2. Views the internal state of the DBF except for the two PRPs F_{TP} and F_{FN}.
3. Submits at most t queries to the DBF.

and then outputs novel false positive with non-negligible probability $\epsilon_{\mathrm{DB}} + \omega\left(\frac{1}{\lambda^c}\right)$. That is, the attacker must be able to construct new inputs that degrade the false positive rate. This threat model is essentially white-box except for the secret PRPs and the attacker's control of the training data.

4 Experiments

The authors simulate the expected false positive rate of DBFs against Naor-Eylon filters with a fixed memory budget. They find that DBFs have a lower false positive rate when the adversary controls only some queries while Naor-Eylon filters have a lower false positive rate when the adversary controls most queries. While DBFs have bounded maximum false positive rate, the expected false positive rate is degraded by tricking the learned model to redirect queries to the "wrong" AMQ filter.

5 Challenging Concepts

I had a tough time with the following concepts

1. Probabilistic polynomial time adversaries
2. Negligible functions
3. Security parameters
4. Pseudo-random permutations

and might need to revisit them later.

References

[1] Allison Bishop and Hayder Tirmazi. “Adversary Resilient Learned Bloom Filters”. In: (2025). arXiv: 2409.06556.

[2] Tim Kraska et al. “The Case for Learned Index Structures”. In: (2018).

[3] Moni Naor and Yogev Eylon. “Bloom Filters in Adversarial Environments”. In: (2019). DOI: 10.1145/3306193.