Quantifying Information Flow

April 10, 2014

1 Belief in Information Flow

This paper shift the measure to quantifying information flow from uncertainty to accuracy. In previous works, information flow was measured by uncertainty, which is characterized by entropy. Entropy returns the number of bits leaked according to the true value of some secret state. The attacker is learning information about the secret, and this entropy is a measure of the reduction in an attacker's uncertainty. The problem with quantifying information flow with uncertainty is that it does not take into account the attacker's initial beliefs about the system. Due to this fact, it is possible for uncertainty to report no information flow when the attacker actually has learned something about secret data.

The new approach in this paper measure accuracy to quantify information flow. Accuracy reasons about the change is an attacker's belief after executing a program. The belief is modeled using distributions, which are mapping from states to probabilities or frequencies. To measure the change in a distribution, the authors use relative entropy as the belief distance. This metric now characterizes the attacker's "surprise" in a result.

The paper also shows how to find the expected information flow of a program. This work is neat since it applies to probabilistic programs.

This approach is similar to a previous paper's notion of channel capacity. The difference is that channel capacity measures the size of a program's output space, but it doesn't assign frequencies to each output. Therefore, channel capacity is like assuming the resulting distribution is uniform.

2 Approximation and Randomization for Quantitative Information-Flow Analysis

This paper does everything to quantify information flow. Specifically, it provides upper and lower bounds on the uncertainty of a program by using sampling, symbolic execution, and abstract interpretation. Their technique is significant because they support data structures and looping over high data.

Their approach works by assuming a uniform distribution over states. This greatly simplifies their analysis since they can simply do counting. Their algorithm in Figure 2 shows how they find confidence intervals by sampling. They use symbolic execution to provide a lower bound on the uncertainty. The execution provides a formula that can be solved to represent possible pre-images of states. This is an under approximation since you can't practically symbolically execute all paths. To compute the upper bound, they perform abstract interpretation which generates an over approximation on the number of output states.

This work also can be used for probabilistic programming since they can just sample to approximate the output distribution.