When in Doubt Throw It Out: Building on Confident Learning for Vulnerability Detection

Artifacts Available
Artifacts Evaluated Reusable
Artifacts Evaluated Functional

Yuanjun Gong · Fabio Massacci

2025 ACM/IEEE 47th International Conference on Software Engineering: New Ideas and Emerging Results

SERICS
SECURITY AND RIGHTS IN THE CYBERSPACE
SEC 4AI4 SEC
Funded by the European Union

What Confident learning is all about

IF dataset not so good → use model to re-label it

- IF Score(x,1) < Confid(Score(x,1)) → label(x,0)
- and so on for 0

BUT

For vulnerabilities, model aren’t good either…

- Wen et al. ICSE’23 → Best precision < 55%
- Croft et al. ICSE’23 → Wrong labels > 28%

Our Idea: when in doubt throw them out

- Relabelling is too risky → too many wrong elements
- Drop elements we are not sure → they are confusing the model

Confidence level = weighted average of the predicted vulns

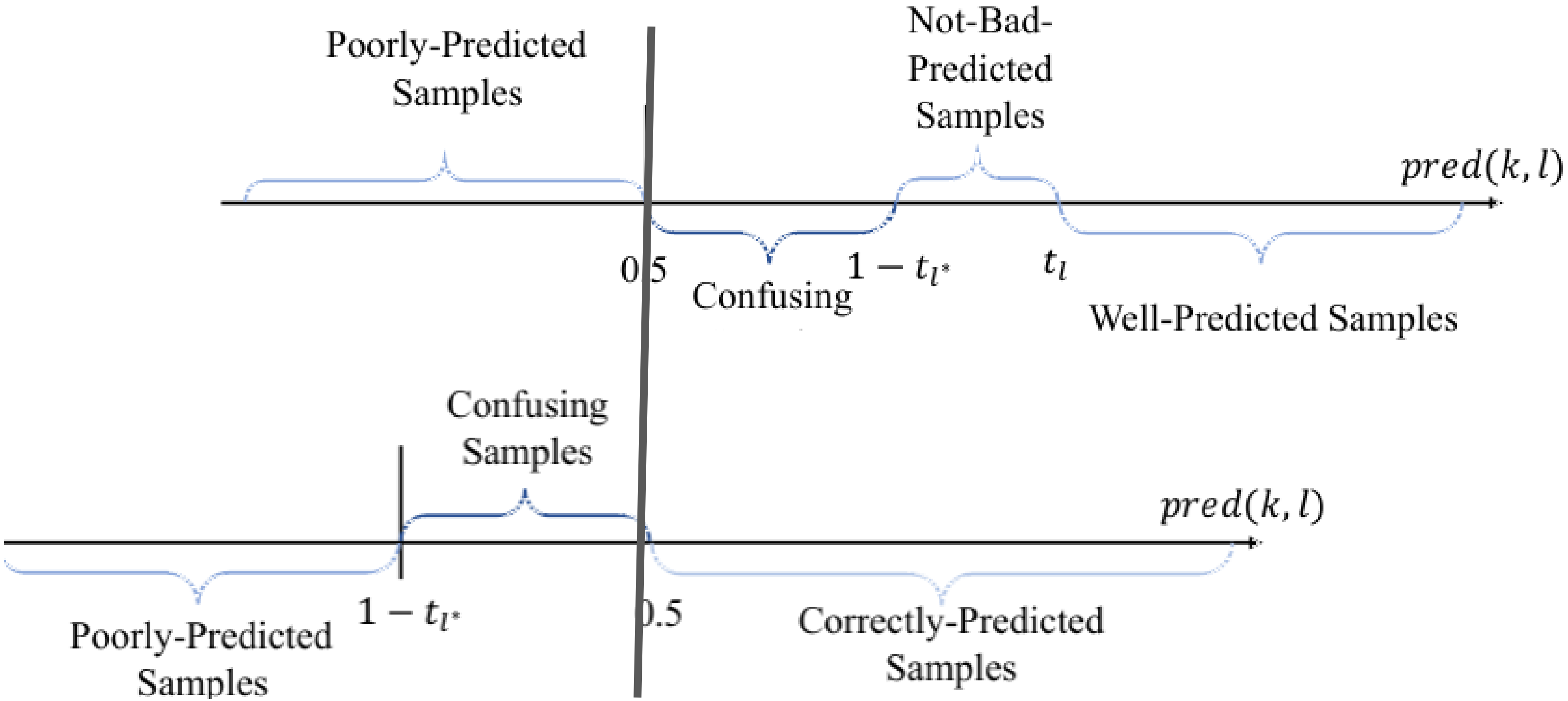
$$t_\ell = \frac{1}{\sum_k label(k,\ell)} \sum_k pred(k,\ell) \cdot label(k,\ell)$$

Well predicted → $pred(k,\ell) > t_\ell > \frac{1}{2}$

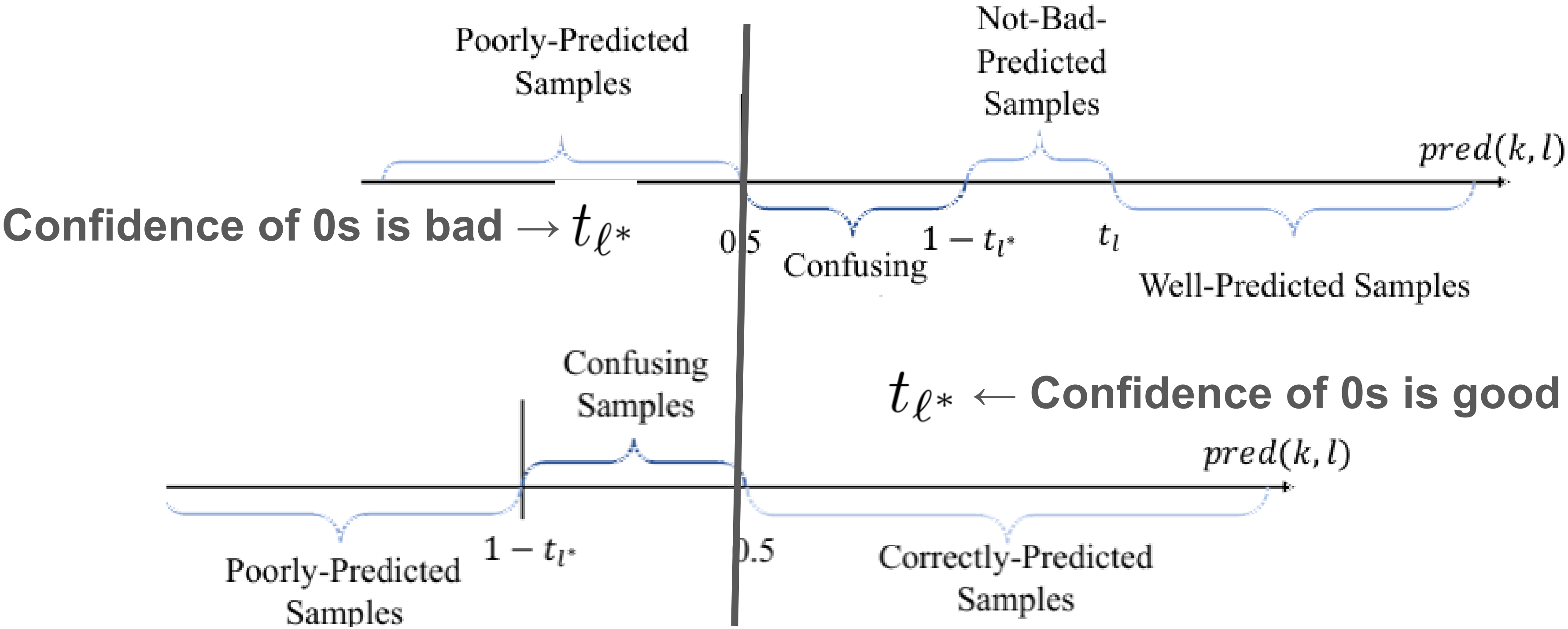
Poorly predicted → $pred(k,\ell) \leq \frac{1}{2}$

The rest → the one confusing the models

Two possible conditions



Two possible conditions → confidence of 0s



And it works….

Running vulnerability dataset with 8K samples (50-50) on a BERT Model

	Data kept	Precision	Recall
Original Dataset	100%	63%	48%
Invert Mislabelled	100%	60%	70%
Remove Mislabelled	72%	59%	77%
Remove Confusing 0	89%	60%	75%
Remove Confusing 1	99%	59%	77%

DROP 1% of datapoints and Recall increases
Without big change on Precision

When in Doubt Throw It Out: Building on Confident Learning for Vulnerability Detection

What Confident learning is all about

IF dataset not so good → use model to re-label it
- IF Score(x,1) < Confid(Score(x,1)) → label(x,0)
- and so on for 0

BUT
For vulnerabilities, model aren't good either…
- Wen et al. ICSE'23 → Best precision < 55%
- Croft et al. ICSE'23 → Wrong labels > 28%

Our Idea: when in doubt throw them out

- Relabelling is too risky → too many wrong elements
- Drop elements we are not sure → they are confusing the model

Confidence level = weighted average of the predicted vulns

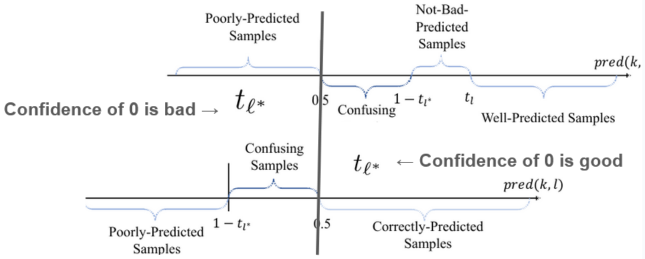
$$t_\ell = \frac{1}{\sum_k label(k,\ell)} \sum_k pred(k,\ell) \cdot label(k,\ell)$$

Well predicted → $pred(k,\ell) > t_\ell > \frac{1}{2}$

Poorly predicted → $pred(k,\ell) \leq \frac{1}{2}$

The rest → the one confusing the models

Two possible conditions → confidence of 0s



And it works….

Running vulnerability dataset with 8K samples (50-50) on a BERT Model

	Data kept	Precision	Recall
Original Dataset	100%	63%	48%
Invert Mislabelled	100%	60%	70%
Remove Mislabelled	72%	59%	77%
Remove Confusing 0	89%	60%	75%
Remove Confusing 1	99%	59%	77%

DROP 1% of datapoint and Recall increases
Without big change on Precision

DOI 10.5281/zenodo.14744047

Yuanjun Gong

Fabio Massacci