

Dependability Evaluation and Enhancing Methods for Transformer Models

ETS PHD FORUM

Transformers models

- Transformers, state-of-the-art of machine learning (ML) models
 - Highly accurate, very complex
 - Many tasks

Transformers models

- Transformers, state-of-the-art of machine learning (ML) models
 - Highly accurate, very complex
 - Many tasks

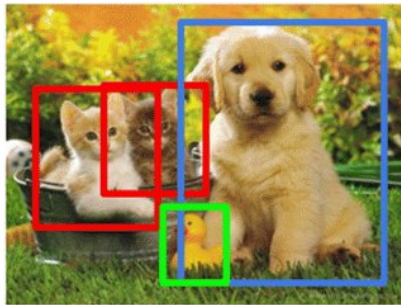
Computer Vision

Classification



CAT

Object Detection



CAT, DOG, DUCK

Transformers models

- Transformers, state-of-the-art of machine learning (ML) models
 - Highly accurate, very complex
 - Many tasks

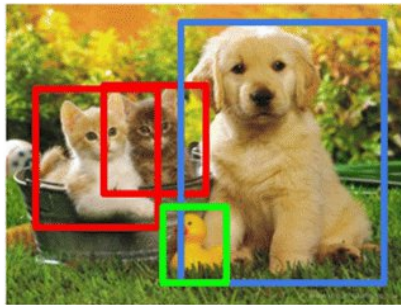
Computer Vision

Classification



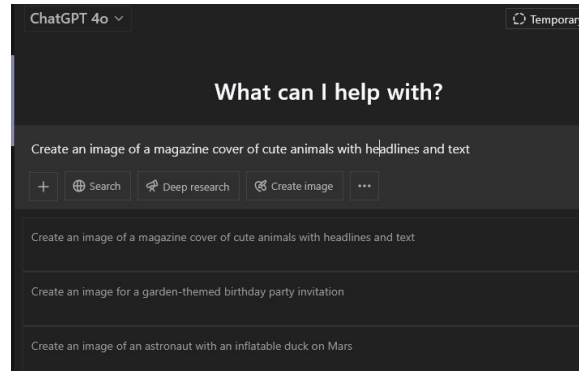
CAT

Object Detection



CAT, DOG, DUCK

NLP



Transformers models

- Transformers, state-of-the-art of machine learning (ML) models
 - Highly accurate, very complex
 - Many tasks

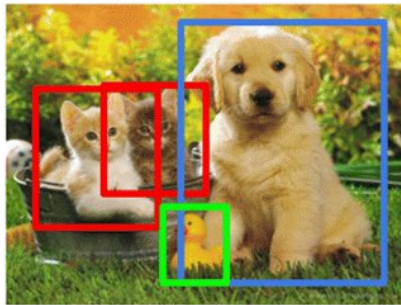
Computer Vision

Classification



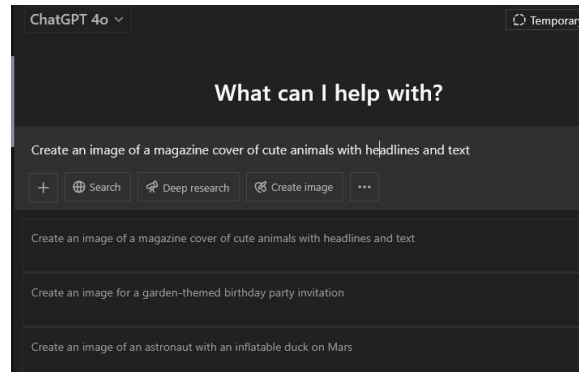
CAT

Object Detection



CAT, DOG, DUCK

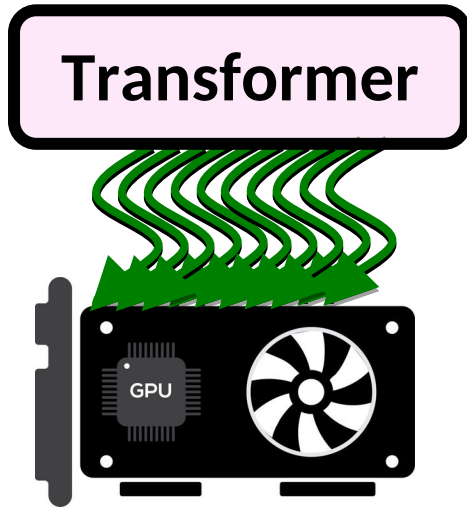
NLP



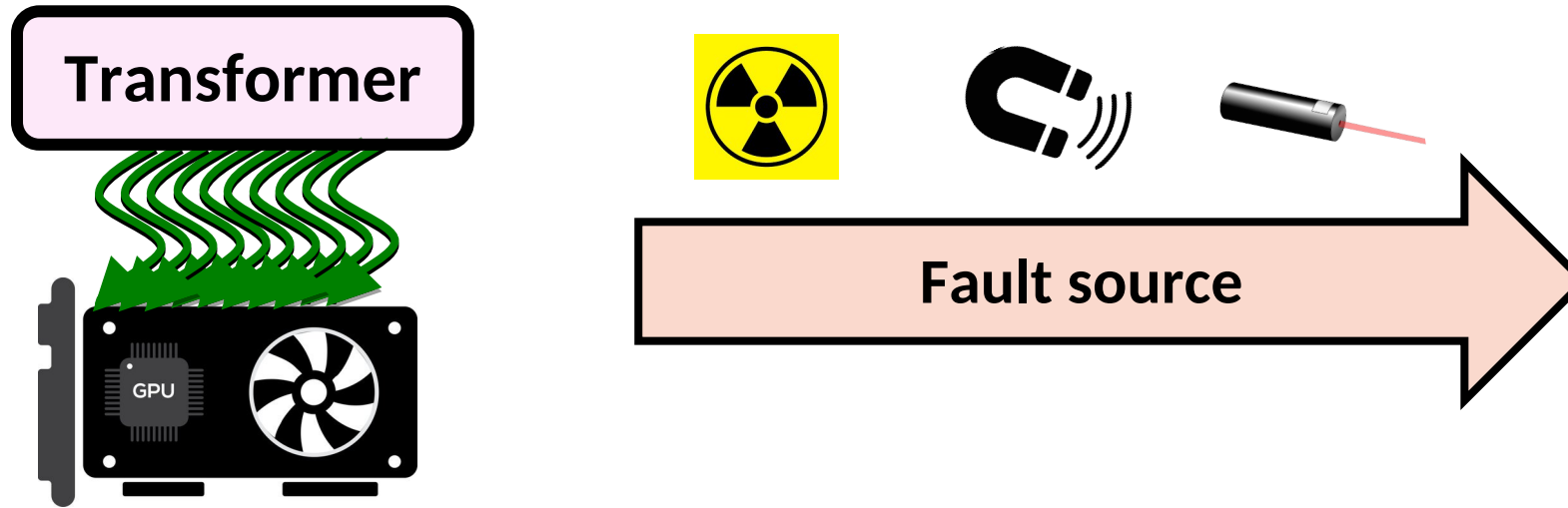
Safety-critical systems



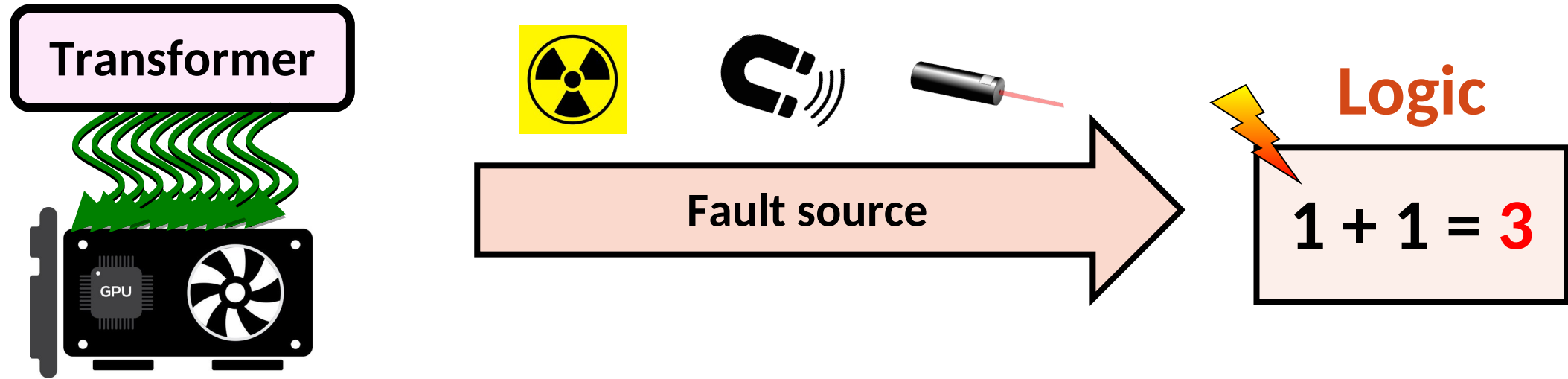
Hardware is vulnerable to faults



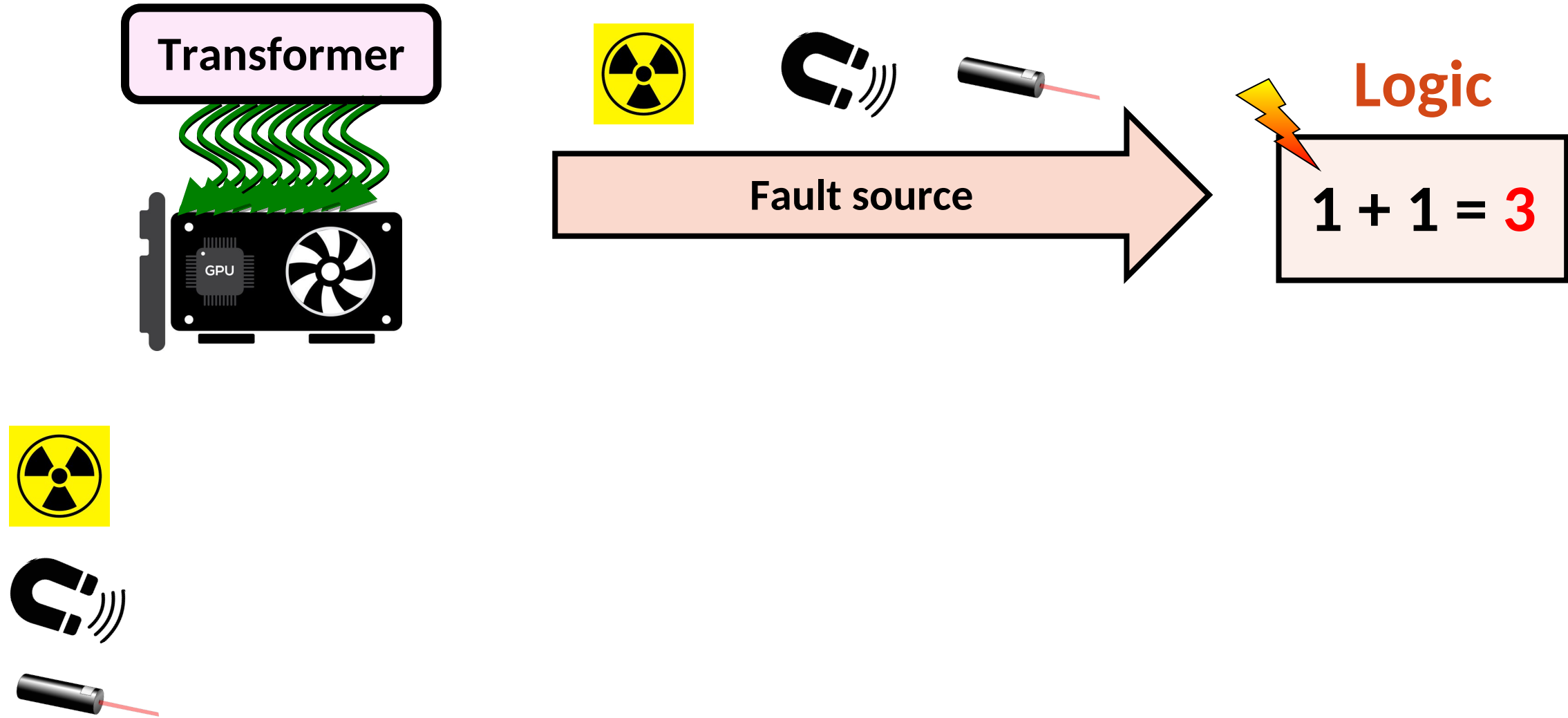
Hardware is vulnerable to faults



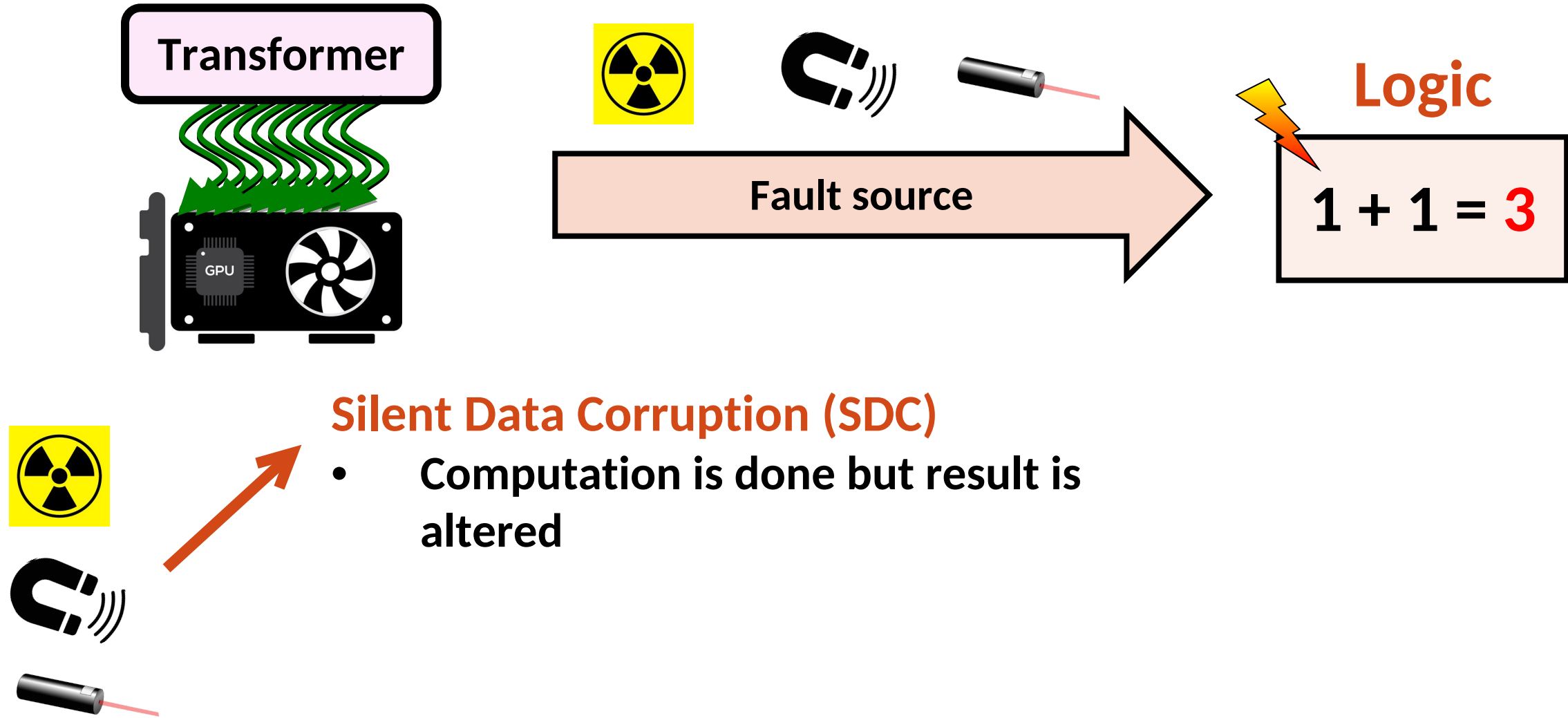
Hardware is vulnerable to faults



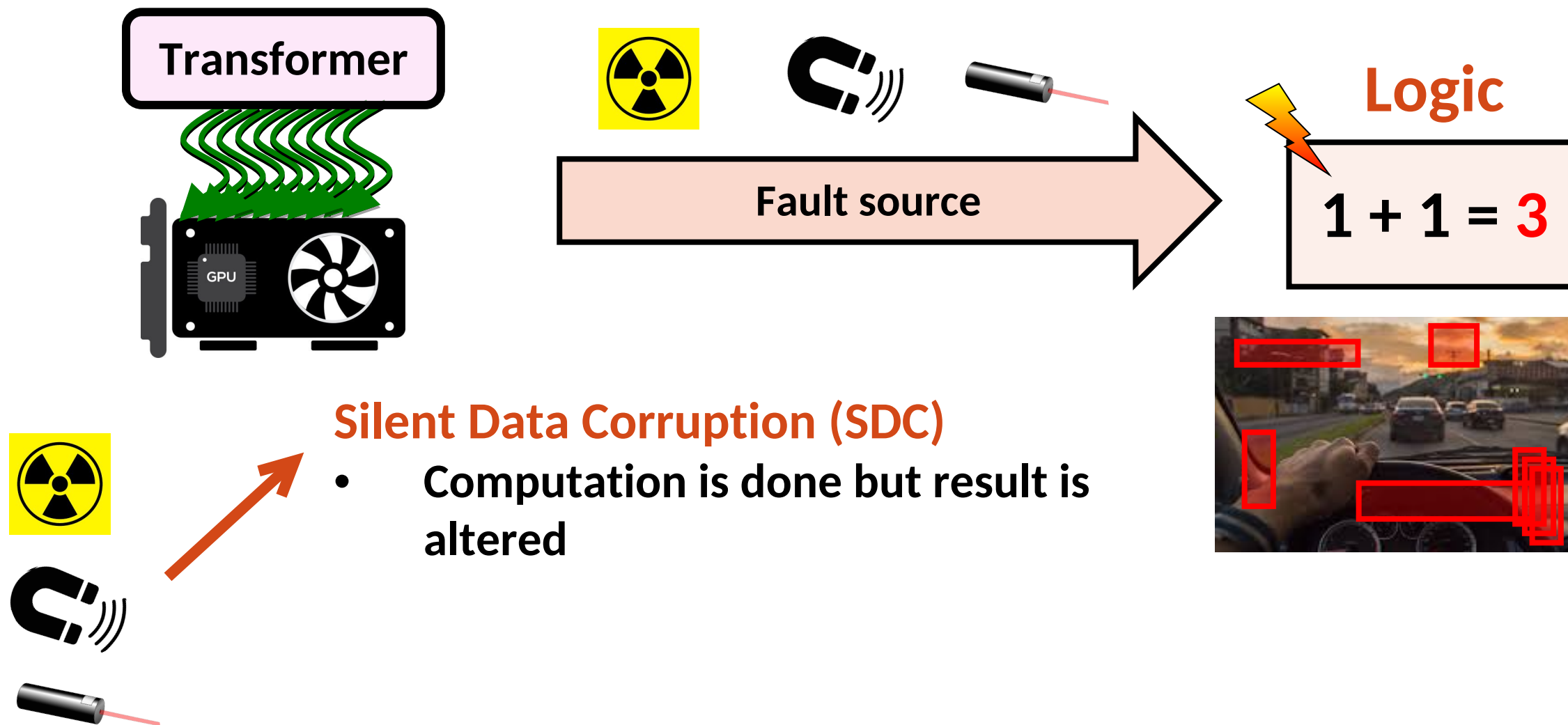
Hardware is vulnerable to faults



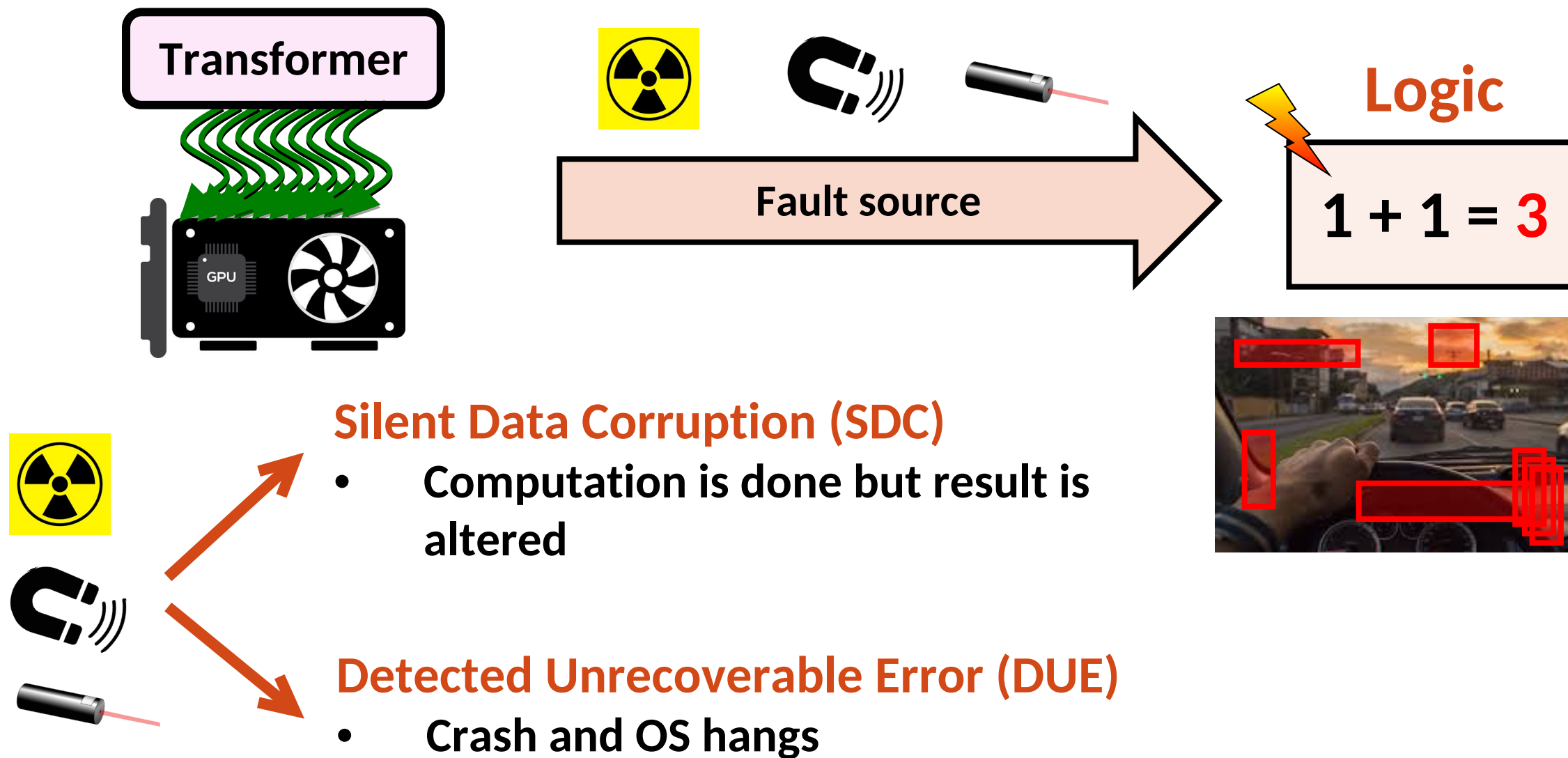
Hardware is vulnerable to faults



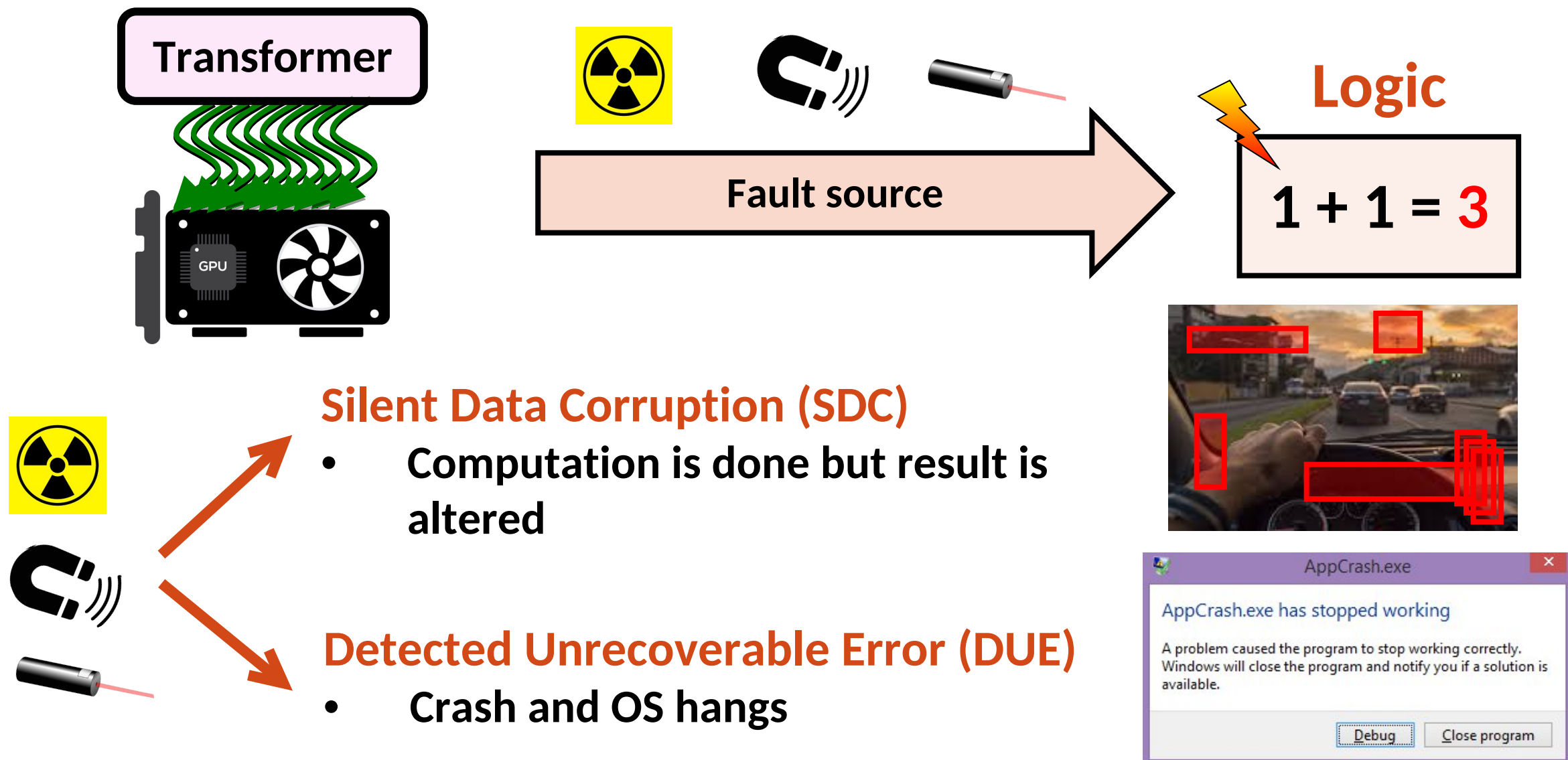
Hardware is vulnerable to faults



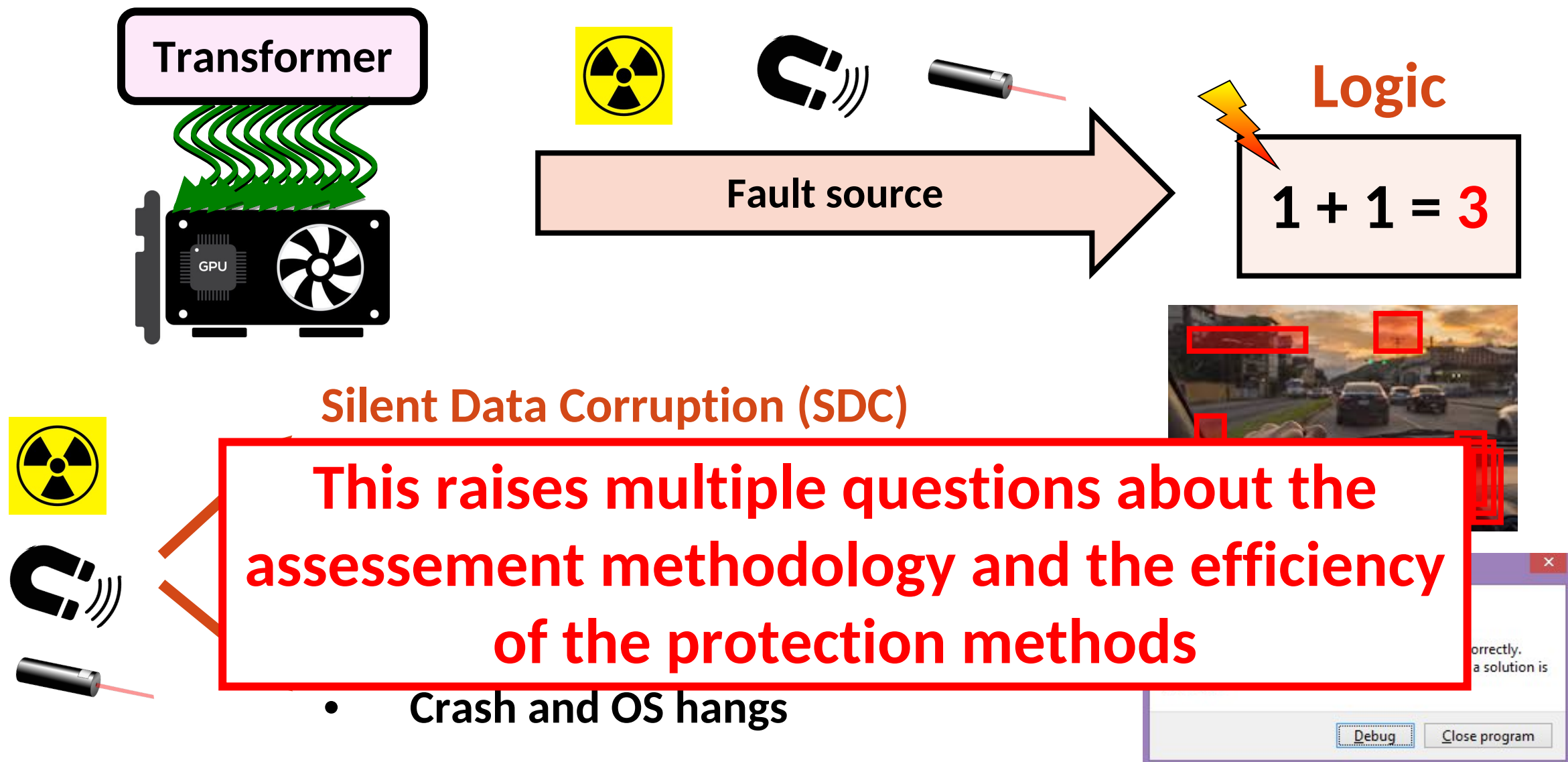
Hardware is vulnerable to faults



Hardware is vulnerable to faults



Hardware is vulnerable to faults



One under-explored variable: inputs

For a dataset of N classes

$Top1$	$Top2$	$Top3$	...	$TopN$
--------	--------	--------	-----	--------

Result: class associated
to $Top1$



One under-explored variable: inputs

For a dataset of N classes

$Top1$	$Top2$	$Top3$	...	$TopN$
--------	--------	--------	-----	--------

Result: class associated
to $Top1$

Example:



One under-explored variable: inputs

For a dataset of N classes

$Top1$	$Top2$	$Top3$	...	$TopN$
--------	--------	--------	-----	--------

Result: class associated
to $Top1$

Example:



Transformer

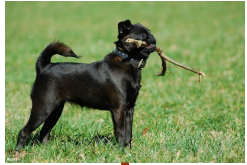
One under-explored variable: inputs

For a dataset of N classes

$Top1$	$Top2$	$Top3$	...	$TopN$
--------	--------	--------	-----	--------

Result: class associated
to $Top1$

Example:



0.98	0.02	0.005	...	0.00..
------	------	-------	-----	--------



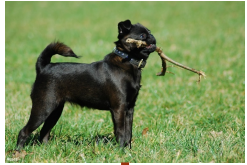
One under-explored variable: inputs

For a dataset of N classes

$Top1$	$Top2$	$Top3$	...	$TopN$
--------	--------	--------	-----	--------

Result: class associated
to $Top1$

Example:



0.98	0.02	0.005	...	0.00..
Dog	Cat	Fish		Car



One under-explored variable: inputs

For a dataset of N classes

$Top1$	$Top2$	$Top3$	...	$TopN$
--------	--------	--------	-----	--------

Result: class associated
to $Top1$

Example:



0.98	0.02	0.005	...	0.00..
Dog	Cat	Fish		Car

We define the confidence as:

$$C = Top1 - Top2$$

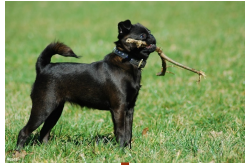
One under-explored variable: inputs

For a dataset of N classes

$Top1$	$Top2$	$Top3$	...	$TopN$
--------	--------	--------	-----	--------

Result: class associated
to $Top1$

Example:



Transformer

0.98	0.02	0.005	...	0.00..
Dog	Cat	Fish		Car

We define the confidence as:

$$C = Top1 - Top2$$



Conf.: 96%,
Class: Dog



Conf.: 0.1%, Class:
Ambulance

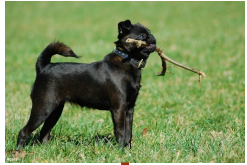
One under-explored variable: inputs

For a dataset of N classes

$Top1$	$Top2$	$Top3$	...	$TopN$
--------	--------	--------	-----	--------

Result: class associated
to $Top1$

Example:



0.98	0.02	0.005	...	0.00..
Dog	Cat	Fish		Car

We define the confidence as:

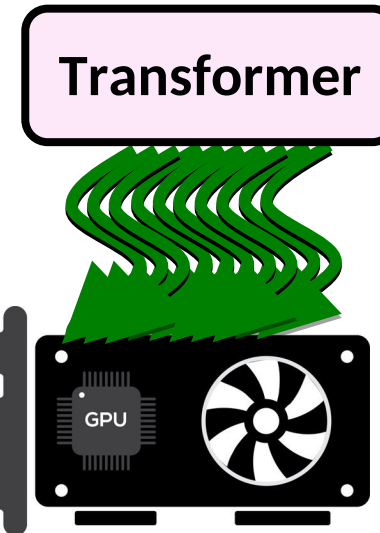
$$C = Top1 - Top2$$



Conf.: 96%,
Class: Dog



Conf.: 0.1%, Class:
Ambulance



One under-explored variable: inputs

For a dataset of N classes

$Top1$	$Top2$	$Top3$	...	$TopN$
--------	--------	--------	-----	--------

Result: class associated
to $Top1$

Example:



0.98	0.02	0.005	...	0.00..
Dog	Cat	Fish		Car

We define the confidence as:

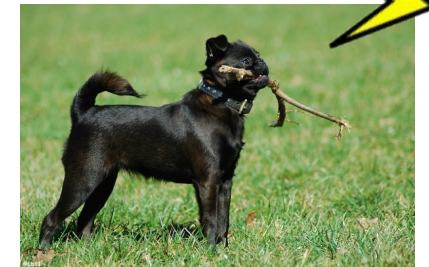
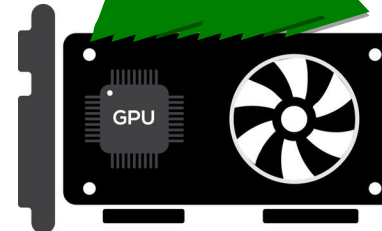
$$C = Top1 - Top2$$



Conf.: 96%,
Class: Dog



Conf.: 0.1%, Class:
Ambulance



Less critical?

Conf.: 80%, Class: Dog



More critical?

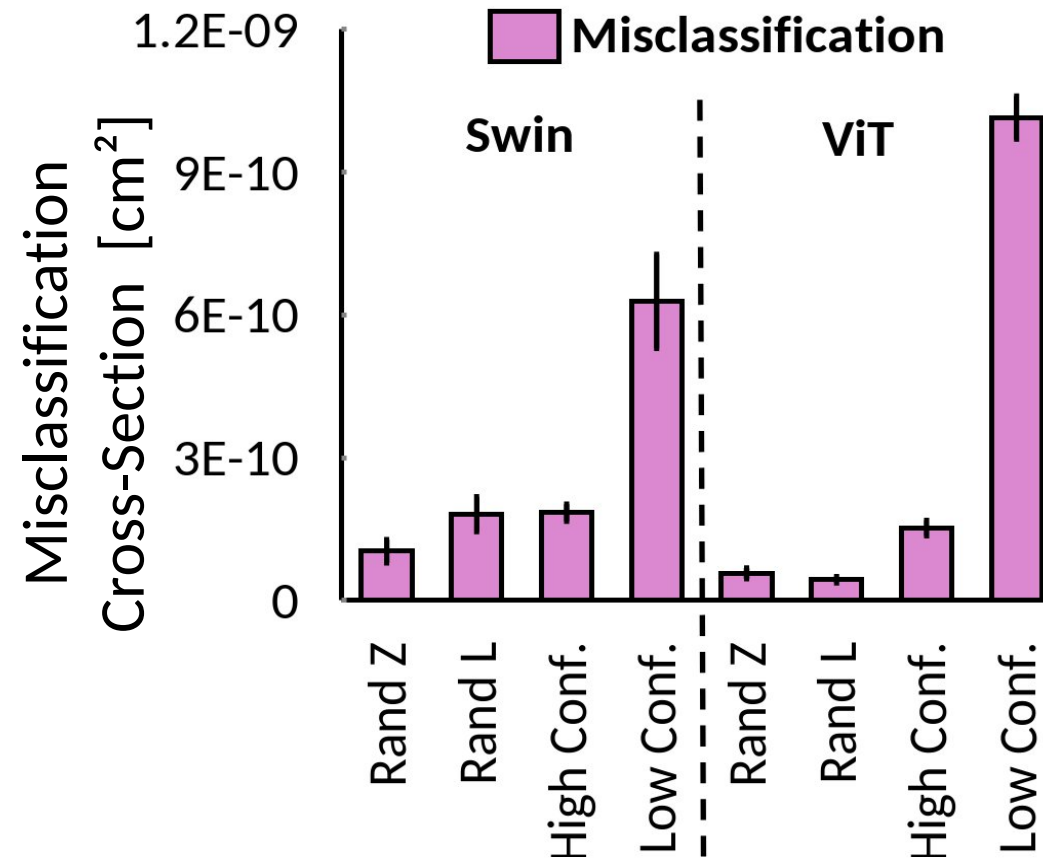
Conf.: 5%, Class: Dog

Results in a nutshell

Results in a nutshell



Neutron beam (unprotected) [1]*

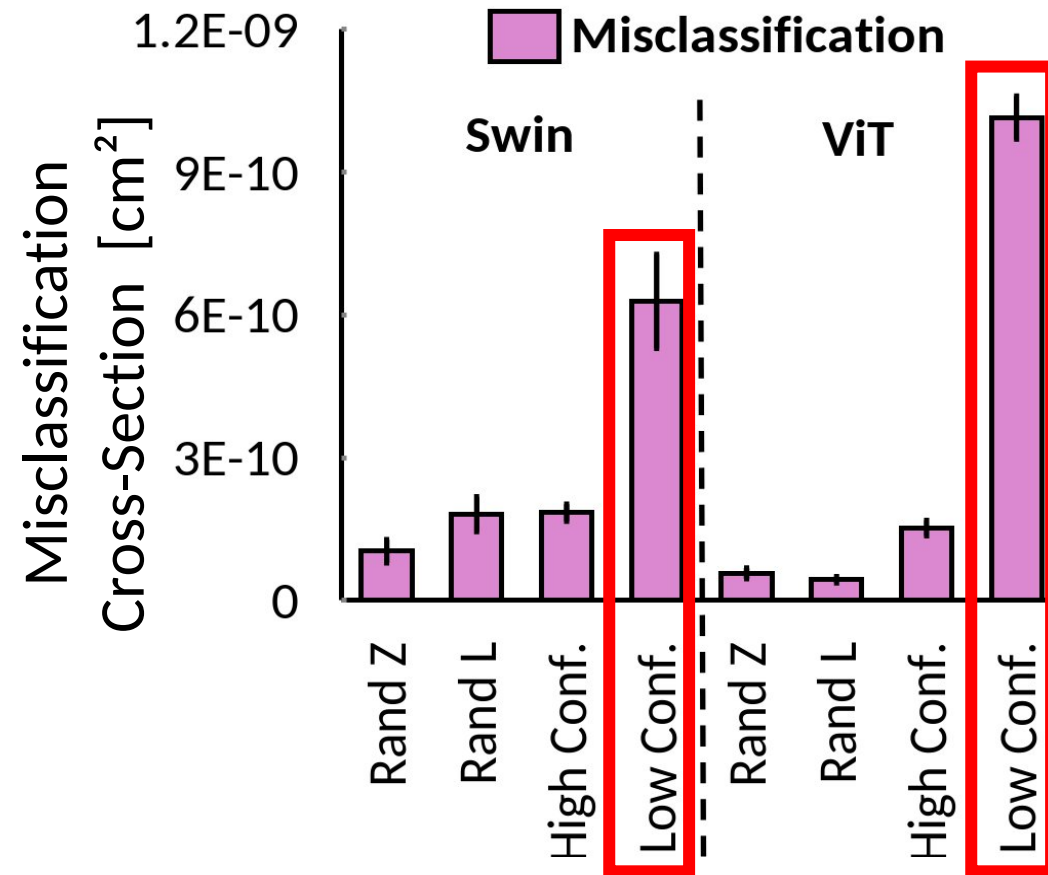


[1] L. Roquet, IEEE TNS, 2026

Results in a nutshell



Neutron beam (unprotected) [1]*

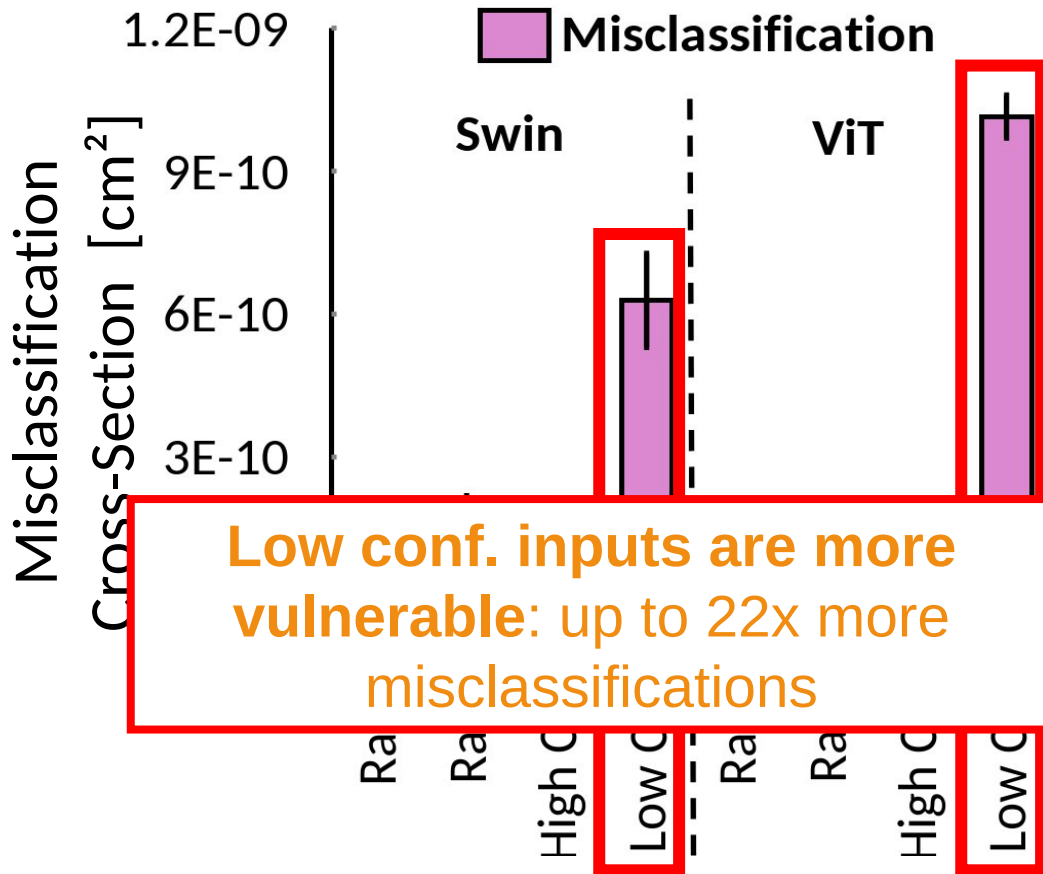


[1] L. Roquet, IEEE TNS, 2026

Results in a nutshell



Neutron beam (unprotected) [1]*

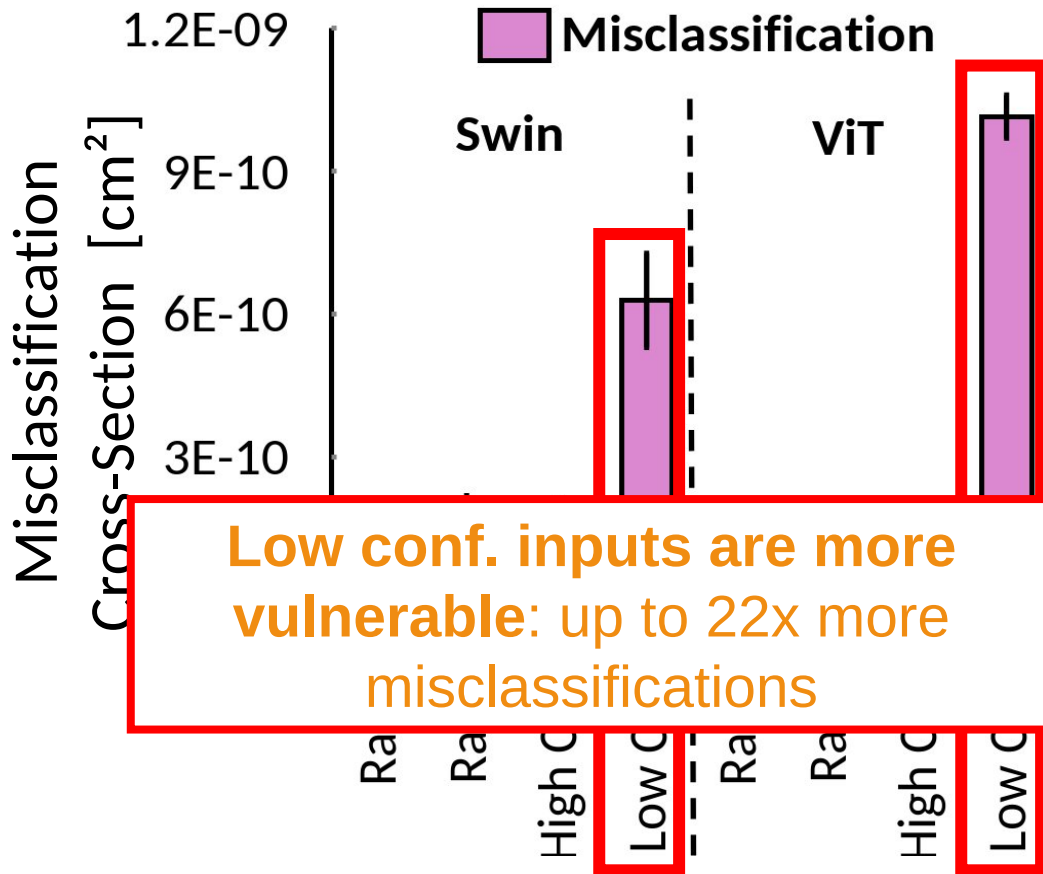


[1] L. Roquet, IEEE TNS, 2026

Results in a nutshell



Neutron beam (unprotected) [1]*



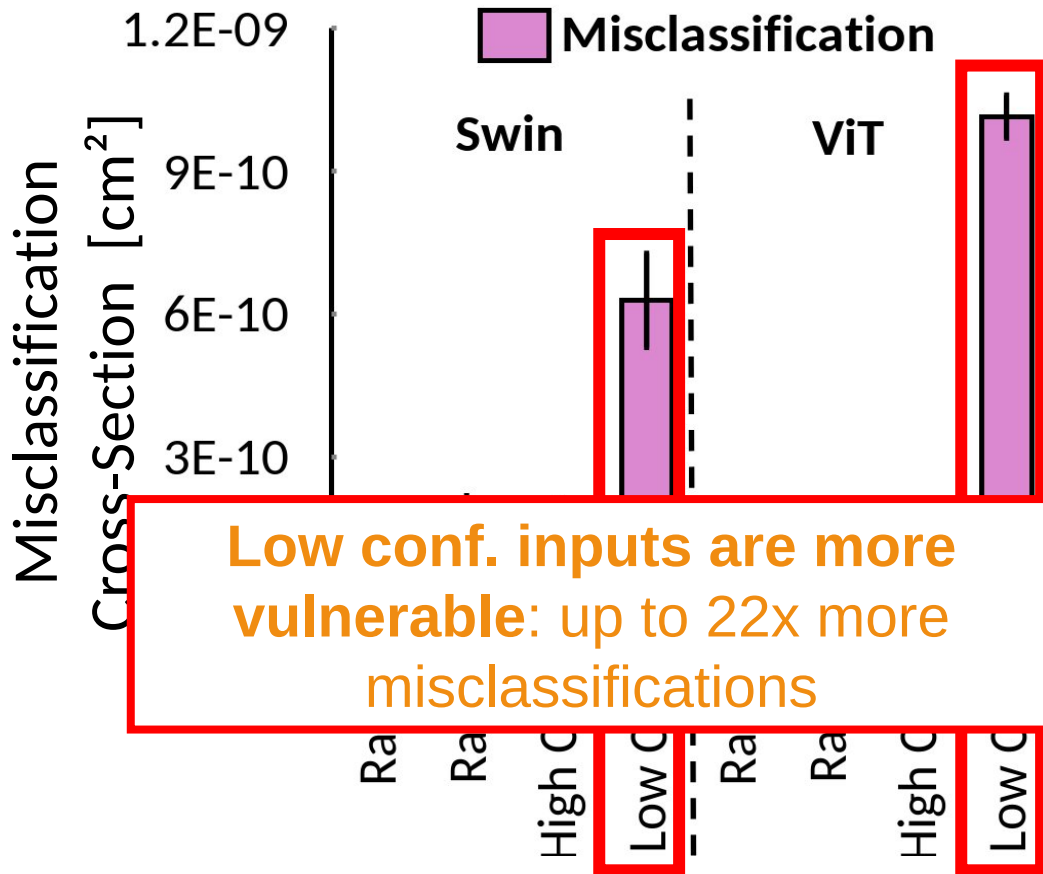
*Same trends are observed by injecting fault at software level

[1] L. Roquet, IEEE TNS, 2026

Results in a nutshell



Neutron beam (unprotected) [1]*



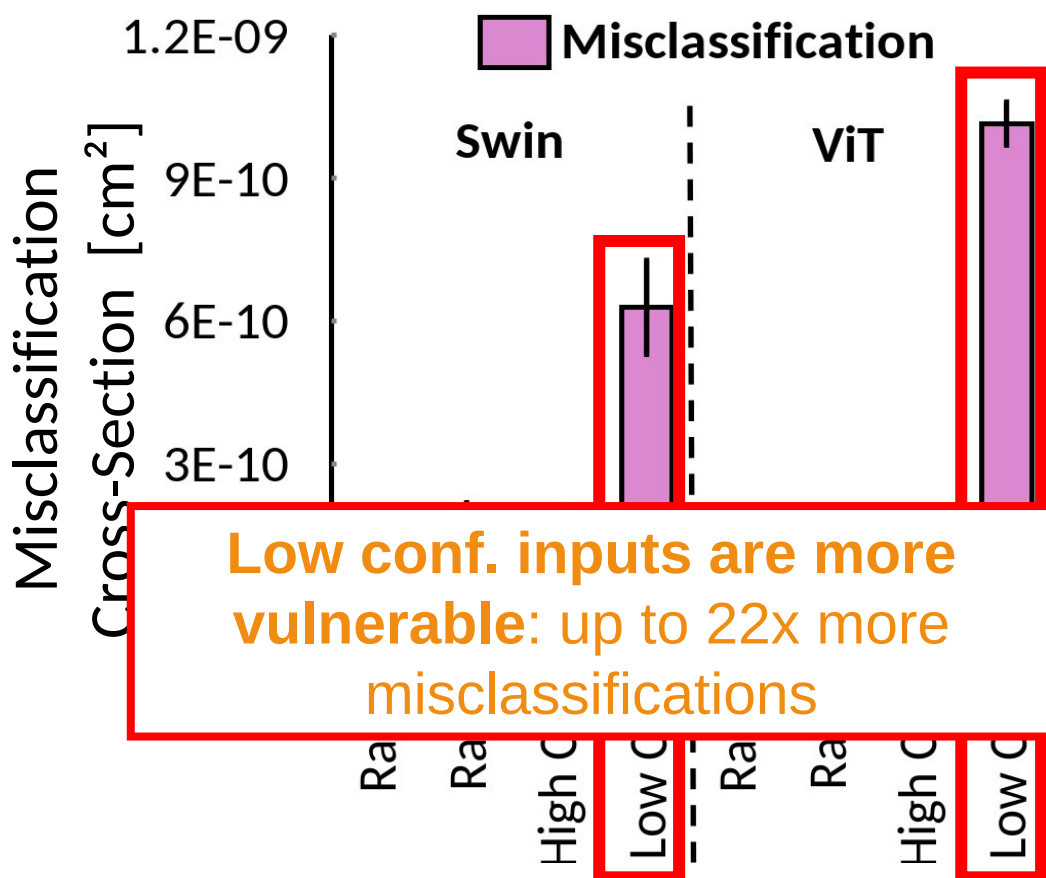
*Same trends are observed by injecting fault at software level

[1] L. Roquet, IEEE TNS, 2026

Results in a nutshell

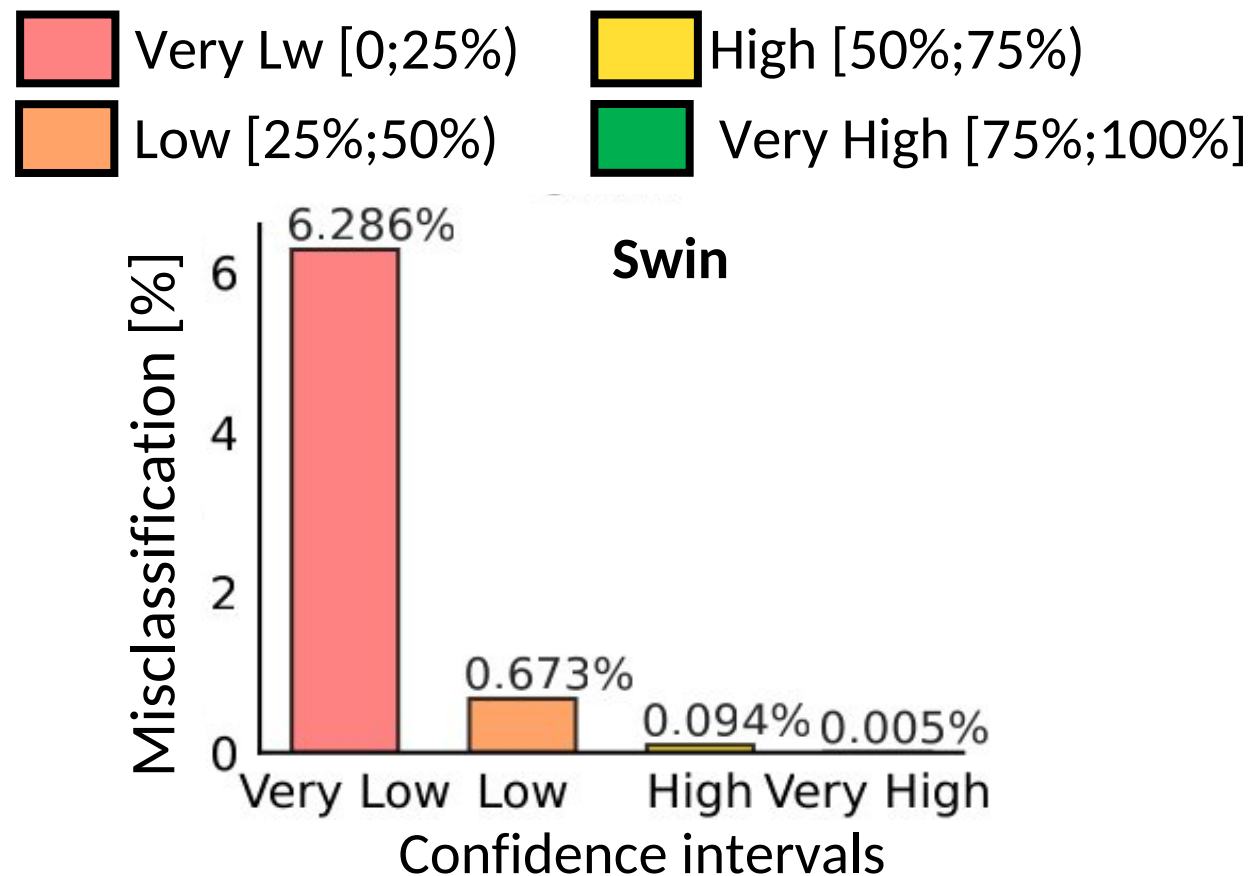


Neutron beam (unprotected) [1]*



* Same trends are observed by injecting fault at software level

Fault simulation (protected) [2]†

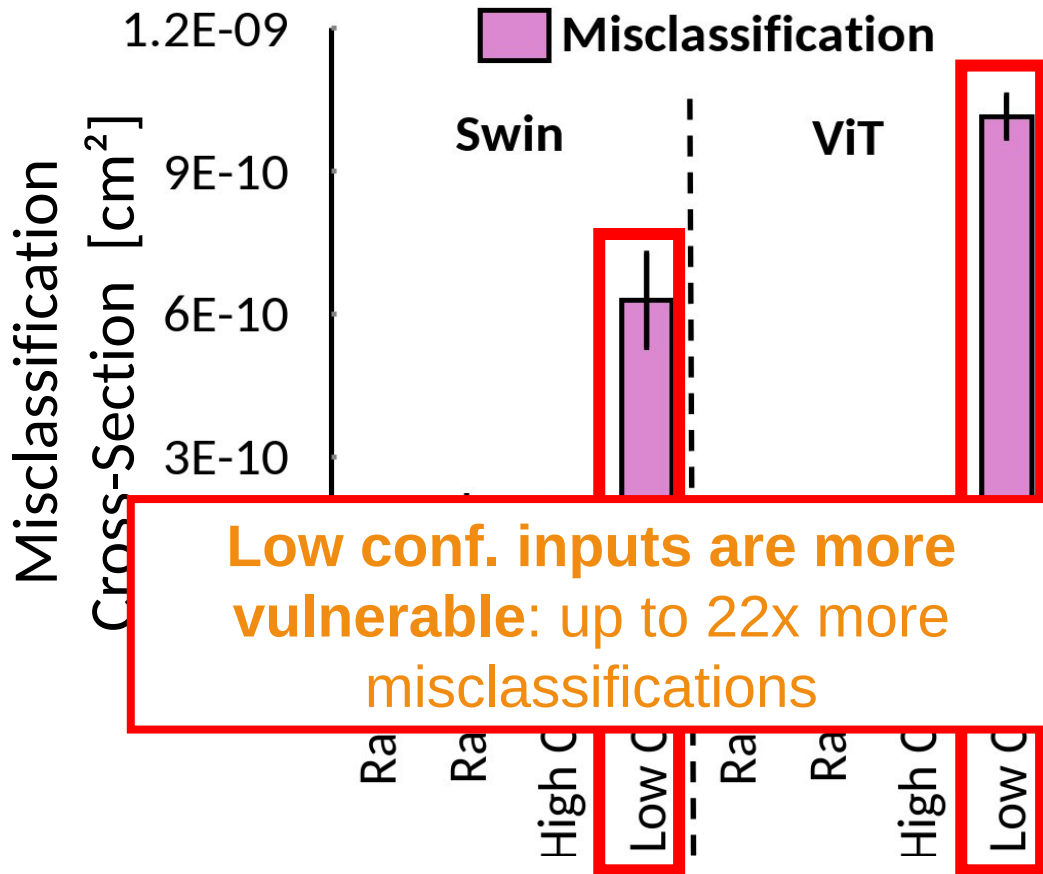


[1] L. Roquet, IEEE TNS, 2026
[2] L. Roquet, IEEE LATS, 2026

Results in a nutshell

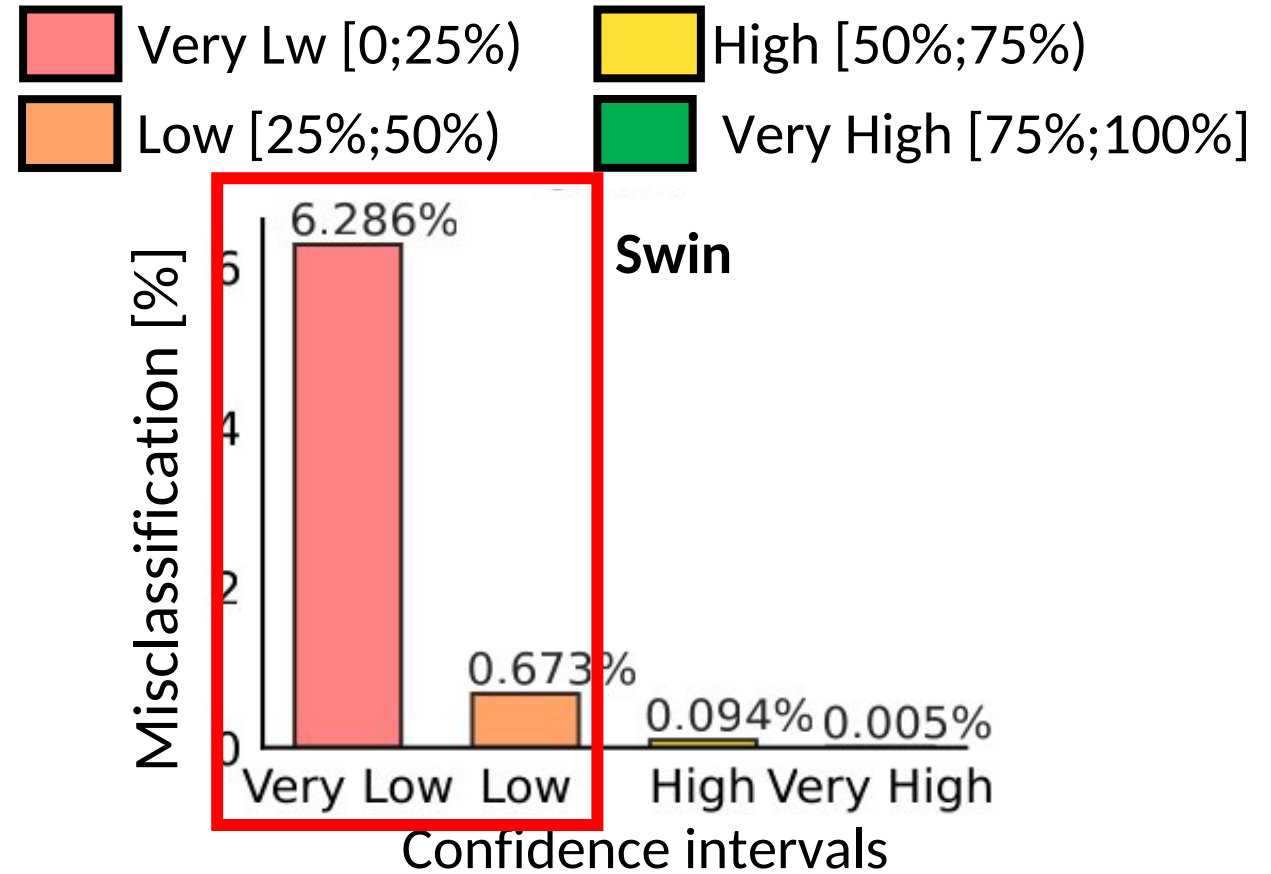


Neutron beam (unprotected) [1]*



* Same trends are observed by injecting fault at software level

Fault simulation (protected) [2]†

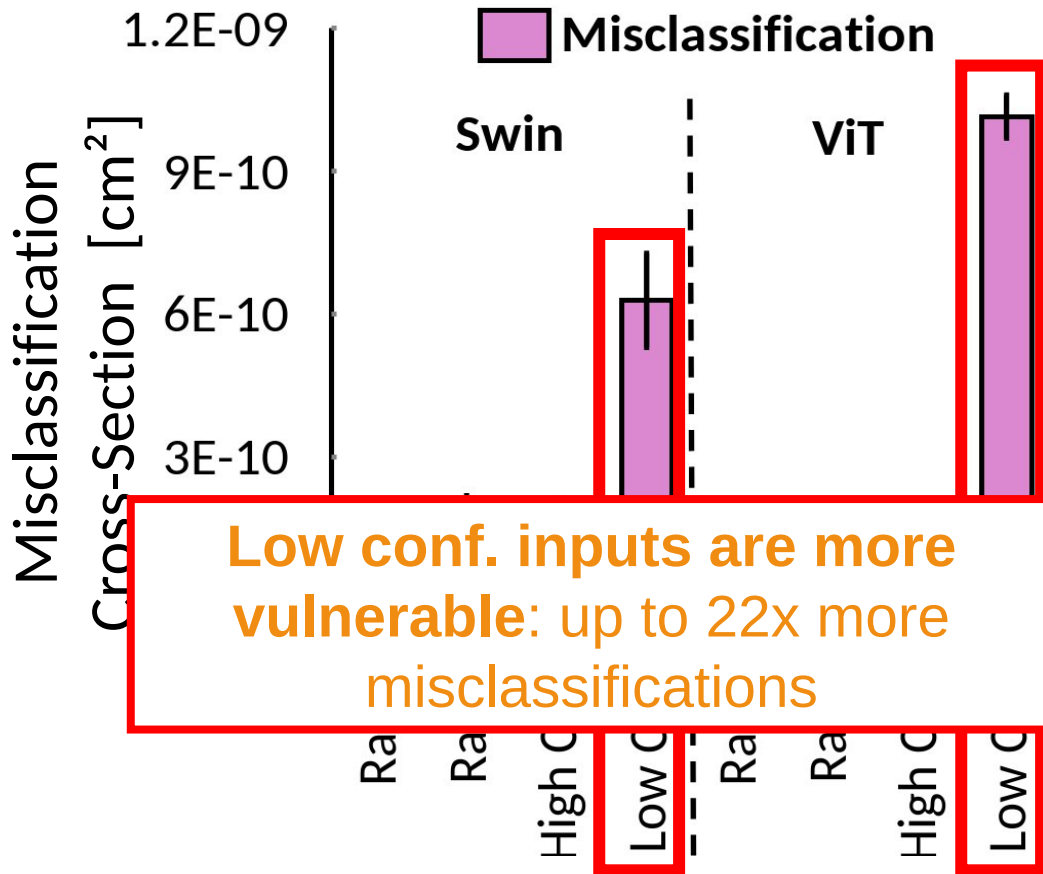


[1] L. Roquet, IEEE TNS, 2026
[2] L. Roquet, IEEE LATS, 2026

Results in a nutshell

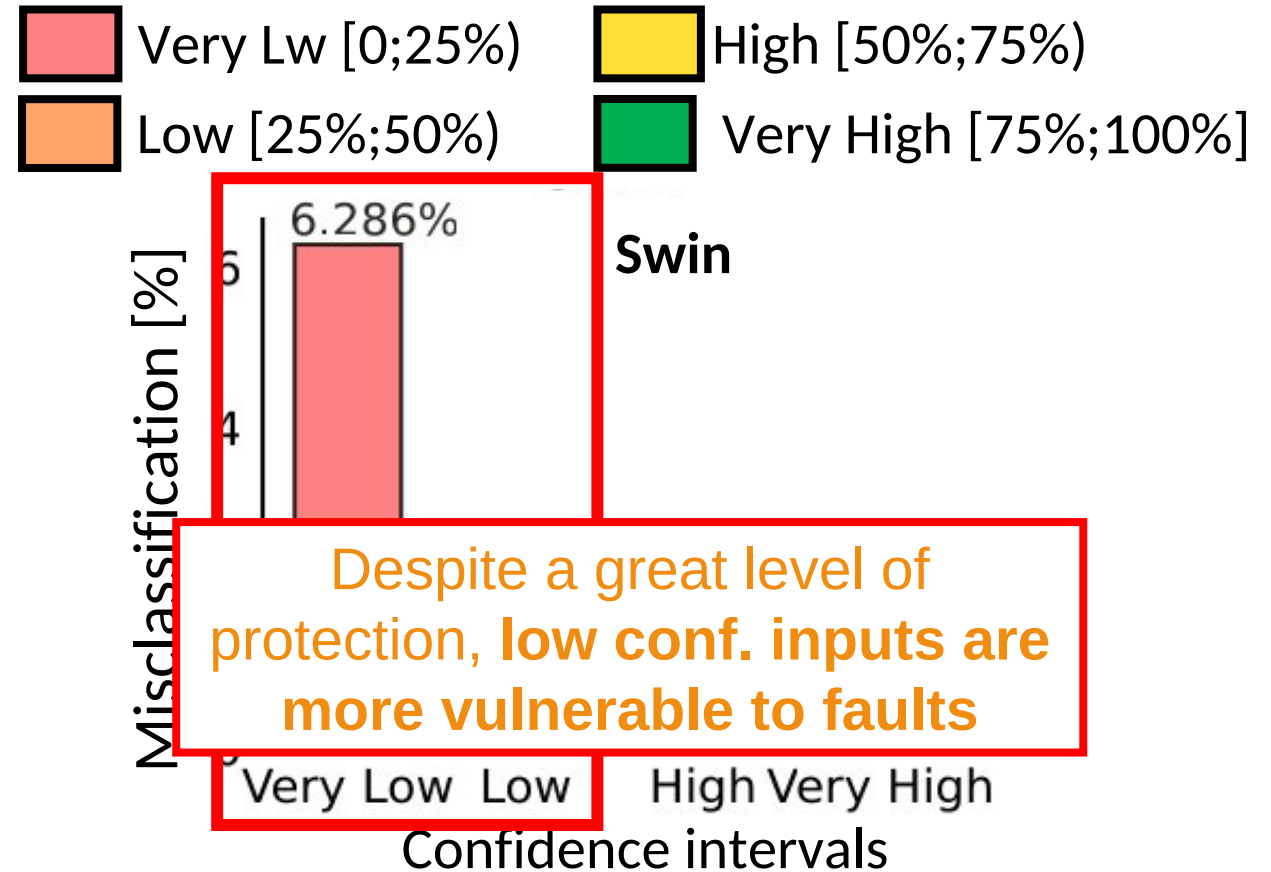


Neutron beam (unprotected) [1]*



* Same trends are observed by injecting fault at software level

Fault simulation (protected) [2]†

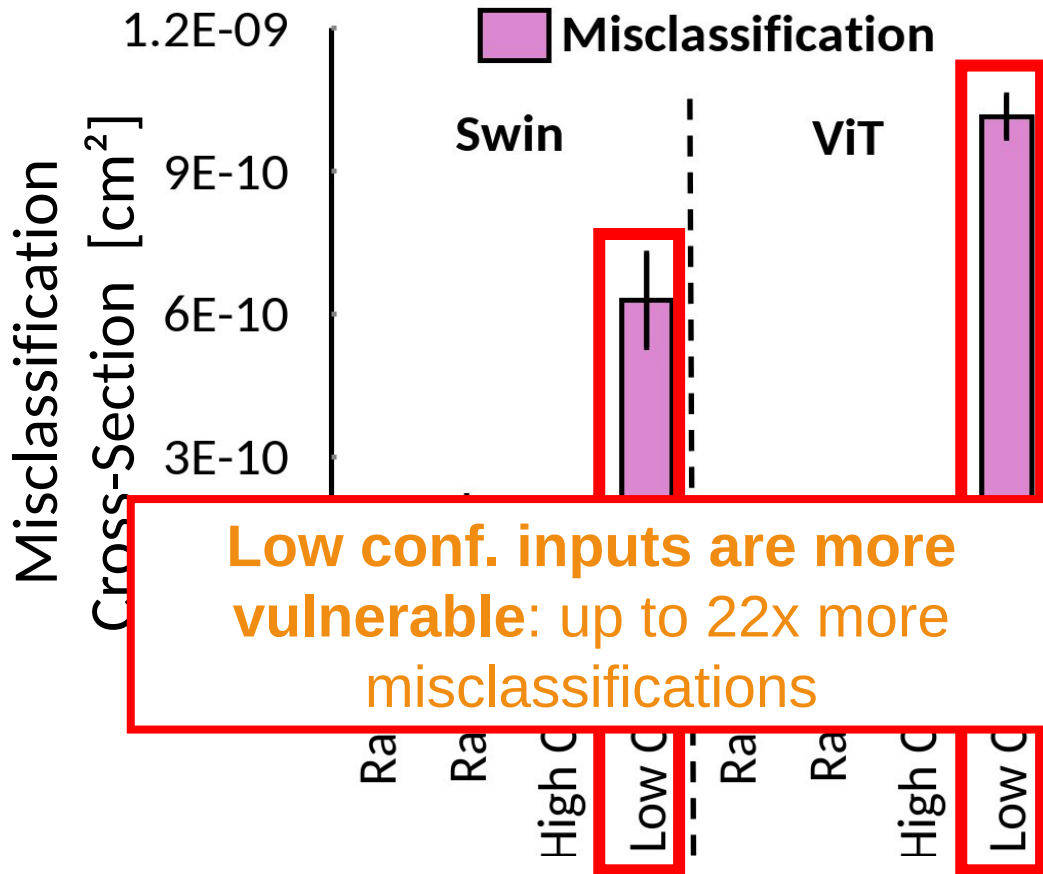


[1] L. Roquet, IEEE TNS, 2026
[2] L. Roquet, IEEE LATS, 2026

Results in a nutshell

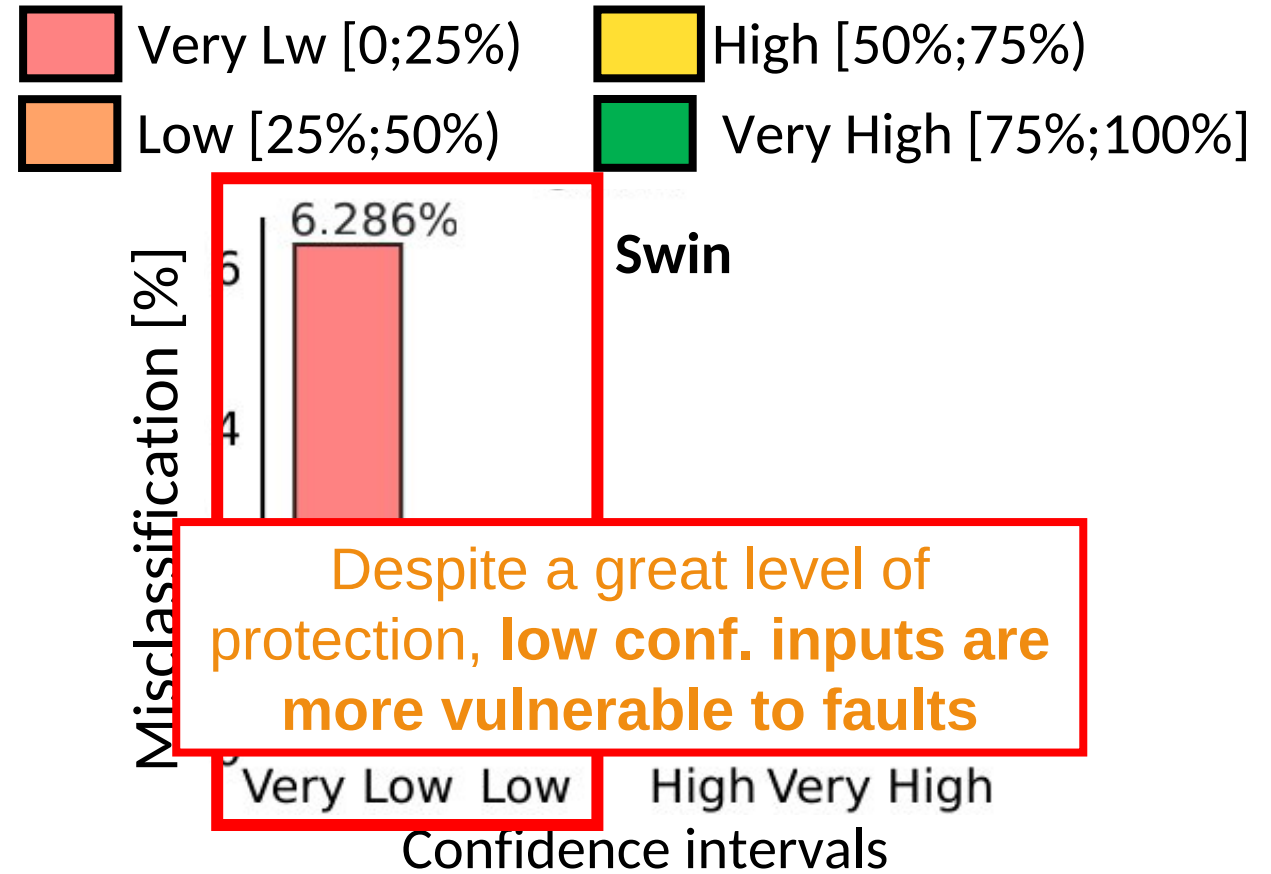


Neutron beam (unprotected) [1]*



*Same trends are observed by injecting fault at software level

Fault simulation (protected) [2]†



†Same trends are observed on 4 different models (CV & NLP)

[1] L. Roquet, IEEE TNS, 2026
[2] L. Roquet, IEEE LATS, 2026

Thank you

Don't hesitate to come see my poster

CONTACT : LUCAS.ROQUET@INRIA.FR

Dependability Evaluation and Enhancing Methods for Transformer Models

IRISA

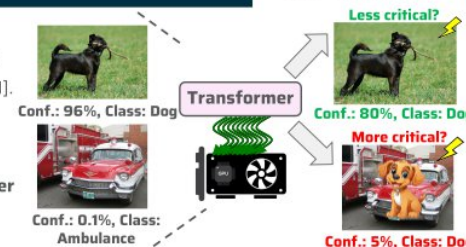
Université
de Rennes

1. Context and Motivation

Transformers running on GPUs are vulnerable to transient faults (i.e. neutrons, protons, etc.) [1].

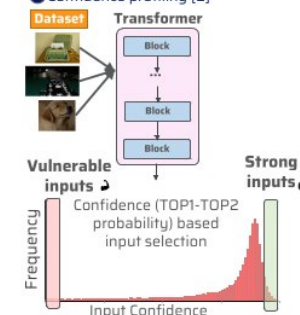
Motivation: Explore how input characteristics affect **misclassification rates**

Goal: Improve the **dependability** of Transformer models

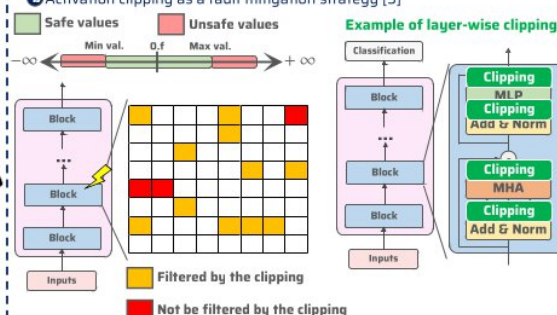


2. Input Selection and Activation Clipping

a Confidence profiling [2]

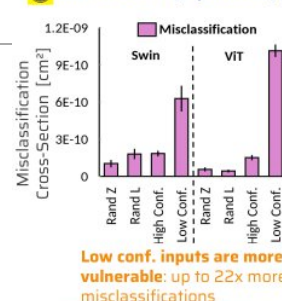


b Activation clipping as a fault mitigation strategy [3]



3. Dependability Evaluation

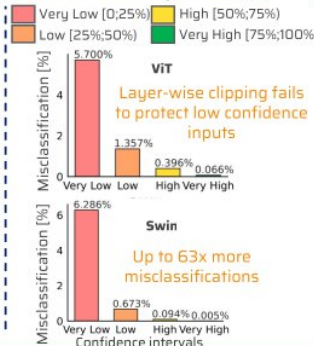
Neutron beam (unprotected) [4]



Lucas ROQUET¹
Fernando FERNANDES
DOS SANTOS¹
Angeliki KRITIKAKOU¹

¹. Univ. Rennes, Inria, IRISA, CNRS

Fault simulation (protected) [5]



Contact
lucas.roquet@inria.fr

References

- [1] P. Rech, IEEE TNS, 2024
- [2] A. Mahmoud et al., IEEE ISSRE, 2021
- [3] Z. Chen et al., IEEE/ACM ISQ, 2021
- [4] L. Roquet et al., IEEE TNS, 2026
- [5] L. Roquet et al., IEEE LATS, 2026

4. Main Takeaways and Future Work

Main Takeaways:

- Similar trends are observed on LLMs
- Misclassification rates are strongly linked to input characteristics
- Despite a strong fault-mitigation strategy, vulnerable inputs are not protected

Future work:

- Develop a tailored fault-mitigation strategy for Transformers based on input characteristics
- Explore security fault-models (laser fault-injection, electromagnetic fault injections, rowhammer, etc.)

Inria

