ISCAS 中国科学院软件研究所学术年会 暨重点实验室科技活动周 2025 第十届

学术论文

The Seeds of the FUTURE Sprout from History: Fuzzing for Unveiling Vulnerabilities in Prospective Deep Learning Libraries

李志远, 吴敬征, 凌祥, 罗天悦, 芮志清, 武延军
lizhiyuan2021, jingzheng08, lingxiang@iscas.ac.cn

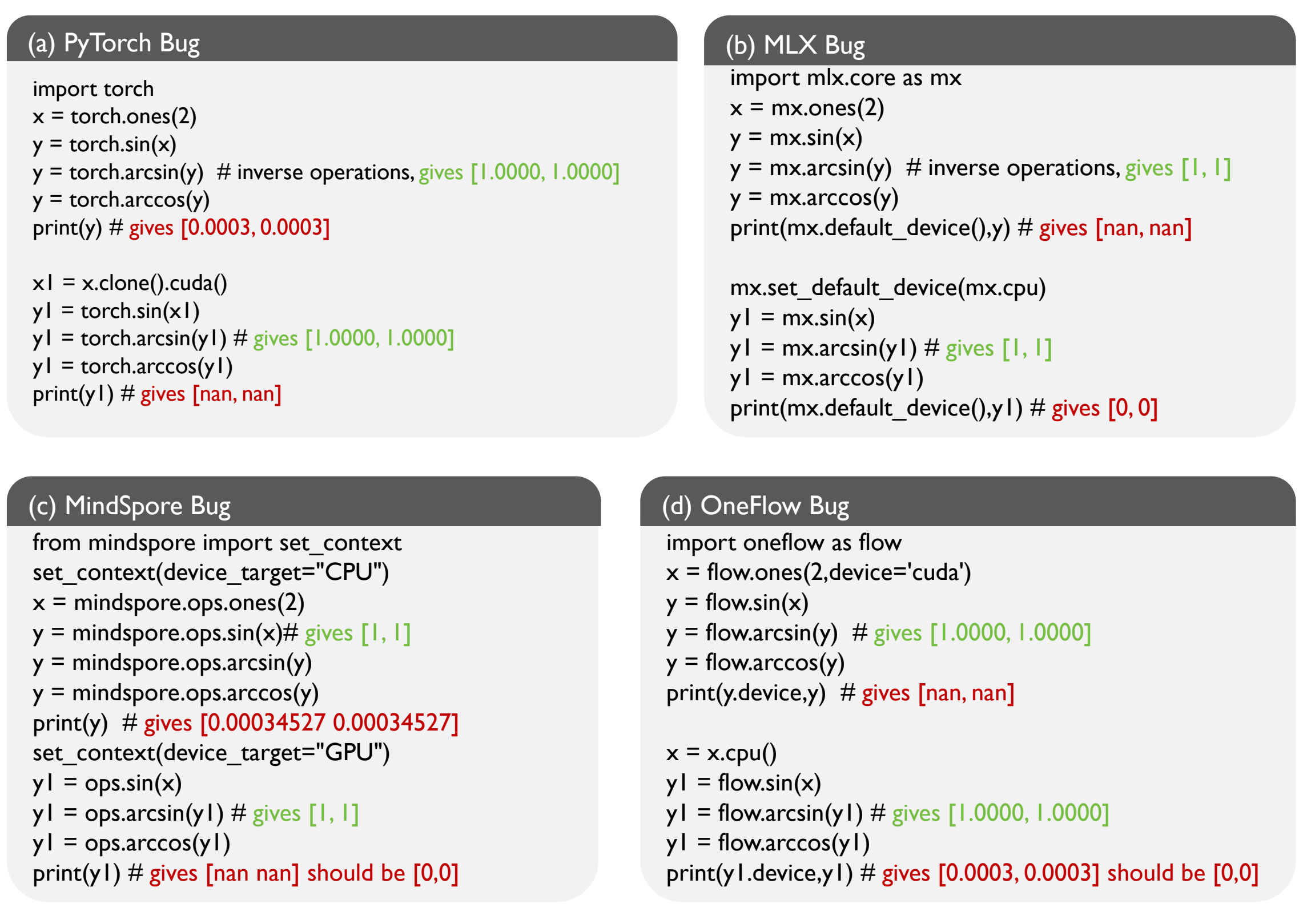
ACM International Conference on Software Engineering (ICSE 2025)
🏆ACM SIGSOFT Distinguished Paper Award

背景 Background

近年来，深度学习技术依赖的人工智能框架安全性日益受到关注。作为检测人工智能框架安全性的关键技术，当前的模糊测试方法存在适用对象局限性、缺乏针对复杂API序列的测试以及大模型训练知识滞后等问题，难以满足开源人工智能框架的安全性检测需求。

动机 Motivation

PyTorch和TensorFlow等传统人工智能框架包含大量历史漏洞信息，这些历史漏洞大多来自真实场景，覆盖了复杂的API序列。在苹果MLX、华为MindSpore以及一流OneFlow等新型框架的文档中明确指出这些新型框架的设计大多受传统框架启发。因此，通过研究团队进行实证研究表明，传统框架的漏洞可能在新型框架上仍然存在或以类似形式存在，故而可以用于新型框架的测试用例生成。



(a) PyTorch Bug

```
import torch
x = torch.ones(2)
y = torch.sin(x)
y = torch.arcsin(y)  # inverse operations, gives [1.0000, 1.0000]
y = torch.arccos(y)
print(y) # gives [0.0003, 0.0003]

x1 = x.clone().cuda()
y1 = torch.sin(x1)
y1 = torch.arcsin(y1) # gives [1.0000, 1.0000]
y1 = torch.arccos(y1)
print(y1) # gives [nan, nan]
```

(b) MLX Bug

```
import mlx.core as mx
x = mx.ones(2)
y = mx.sin(x)
y = mx.arcsin(y)  # inverse operations, gives [1, 1]
y = mx.arccos(y)
print(mx.default_device(),y) # gives [nan, nan]

mx.set_default_device(mx.cpu)
y1 = mx.sin(x)
y1 = mx.arcsin(y1) # gives [1, 1]
y1 = mx.arccos(y1)
print(mx.default_device(),y1) # gives [0, 0]
```

(c) MindSpore Bug

```
from mindspore import set_context
set_context(device_target="CPU")
x = mindspore.ops.ones(2)
y = mindspore.ops.sin(x)# gives [1, 1]
y = mindspore.ops.arcsin(y)
y = mindspore.ops.arccos(y)
print(y)  # gives [0.00034527 0.00034527]
set_context(device_target="GPU")
y1 = ops.sin(x)
y1 = ops.arcsin(y1) # gives [1, 1]
y1 = ops.arccos(y1)
print(y1) # gives [nan nan] should be [0,0]
```

(d) OneFlow Bug

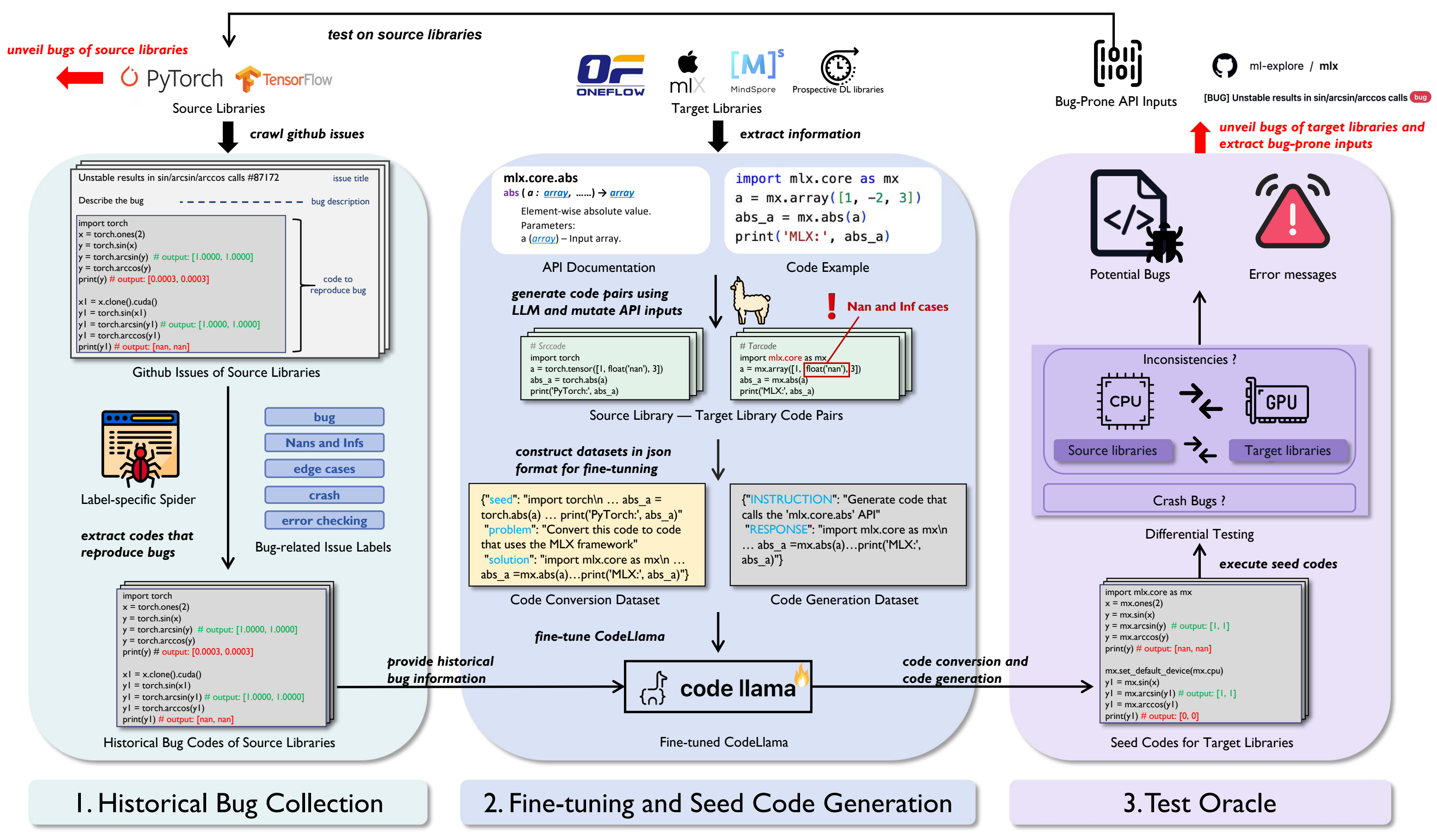
```
import oneflow as flow
x = flow.ones(2,device='cuda')
y = flow.sin(x)
y = flow.arcsin(y)  # gives [1.0000, 1.0000]
y = flow.arccos(y)
print(y.device,y)  # gives [nan, nan]

x = x.cpu()
y1 = flow.sin(x)
y1 = flow.arcsin(y1) # gives [1.0000, 1.0000]
y1 = flow.arccos(y1)
print(y1.device,y1) # gives [0.0003, 0.0003] should be [0,0]
```

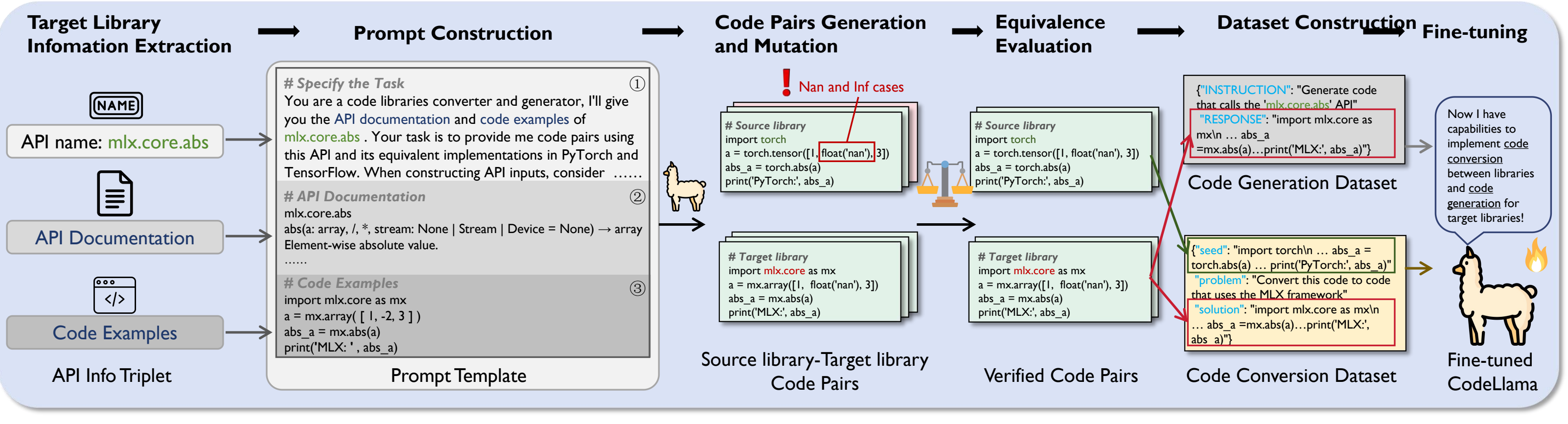
PyTorch某漏洞在MLX, MindSpore, OneFlow仍存在

方法 Methodology

针对上述需求和动机，研究团队创新性地提出了面向开源人工智能框架的模糊测试方法FUTURE。该方法利用新型源AI框架的文档说明与代码示例构建微调数据集，然后通过LoRA技术实现对大语言模型的微调，并利用微调后模型构建框架间的代码映射关系，将PyTorch、TensorFlow等传统开源框架中的大量历史漏洞转换为新型框架的测试用例，通过差分测试运行测试用例并捕捉其中的异常行为，从而实现对开源人工智能框架安全性的有效检测。



FUTURE框架图

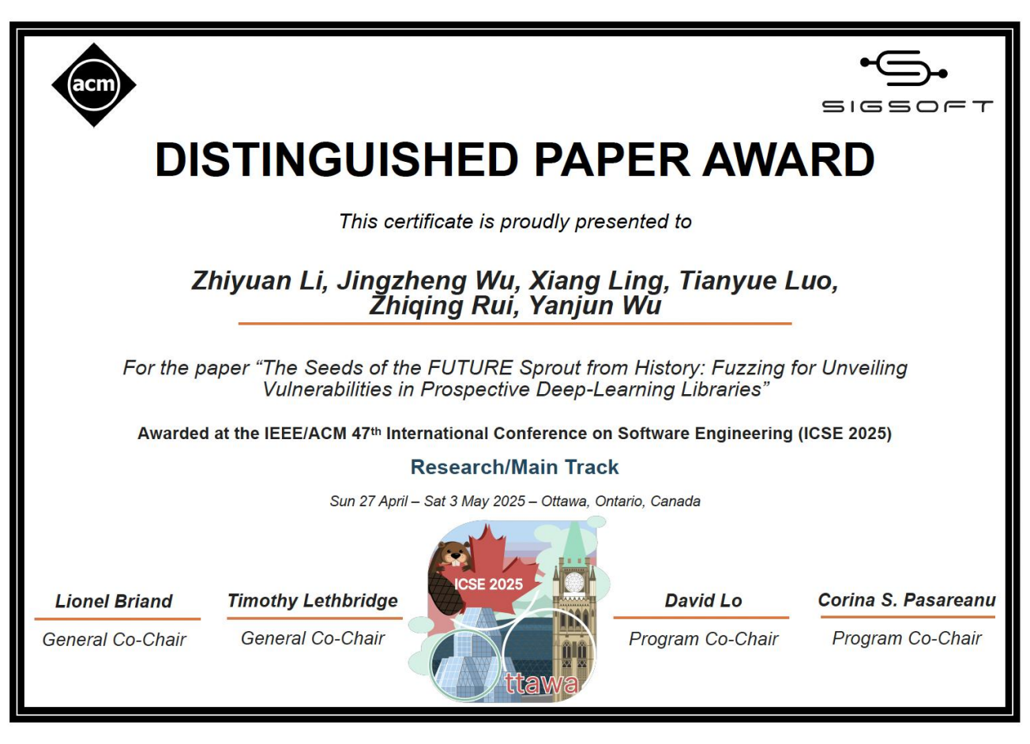


```
# Specify the Task ①
You are a code libraries converter and generator, I'll give you the API documentation and code examples of mlx.core.abs . Your task is to provide me code pairs using this API and its equivalent implementations in PyTorch and TensorFlow. When constructing API inputs, consider ......
# API Documentation ②
mlx.core.abs
abs(a: array, /, *, stream: None | Stream | Device = None) → array
Element-wise absolute value.
......
# Code Examples ③
import mlx.core as mx
a = mx.array( [ 1, -2, 3 ] )
abs_a = mx.abs(a)
print('MLX: ' , abs_a)
```

数据集构建与大模型微调过程

实验结果 Experiment Results

FUTURE在开源人工智能框架MindSpore、MLX、OneFlow、PyTorch进行了测试。结果显示，FUTURE累计发现新缺陷155个，其中10个缺陷获得CVE编号和4个缺陷获得CNVD编号。此外，该方法还发现了20个可优化问题与12个文档问题。该论文获得ICSE 2025杰出论文奖Distinguished Paper Award。



acm SIGSOFT

DISTINGUISHED PAPER AWARD

This certificate is proudly presented to

Zhiyuan Li, Jingzheng Wu, Xiang Ling, Tianyue Luo, Zhiqing Rui, Yanjun Wu

For the paper "The Seeds of the FUTURE Sprout from History: Fuzzing for Unveiling Vulnerabilities in Prospective Deep-Learning Libraries"

Awarded at the IEEE/ACM 47th International Conference on Software Engineering (ICSE 2025)

Research/Main Track

Sun 27 April – Sat 3 May 2025 – Ottawa, Ontario, Canada

Lionel Briand	Timothy Lethbridge	David Lo	Corina S. Pasareanu
General Co-Chair	General Co-Chair	Program Co-Chair	Program Co-Chair

GitHub仓库 论文链接

This paper is supported by

Open Source Map