Beijing Forest Studio
北京理工大学信息系统及安全对抗实验中心

Weakness Identification of Binary Program Function of Pseudo-code by Incorporating Structure and Sequence Information with Attention-Residual Connections

PhD: ABDULRAHMAN CHUKKOL 韩小初

老师：Prof. LUO SENLIN

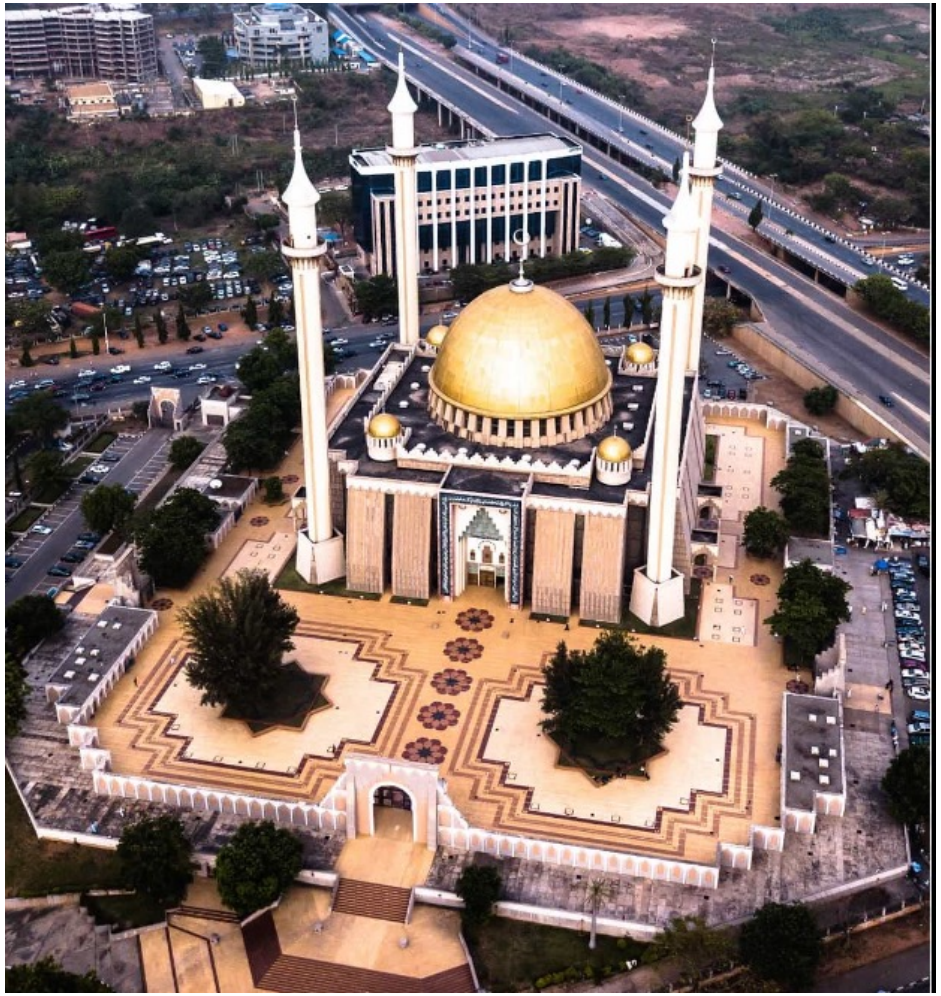
2023年07月02日

Contents

- Self Introduction
- Research objectives and analysis
- Research background and significance
- Research trend
- Research content
- Key technologies and solutions
  - Weakness Identification of Binary Program Function
- Experimental system
- Thank You

Self Introduction

- Born in Nigeria – West Africa.
- The most populous black nation in the world.
- The largest Oil producers in Africa.
- Has the reachest black man in the world.
- Has the largest Entertainment industry.



Self Introduction

- Attended primary and secondary schools in Nigeria.
- 2010-2012: Diploma in Information Technology and communication/Networks and Cyber security from Informatics Academy Singapore.

- 2013-2015: BSc Information Technology from Nims University Jaipur, India.

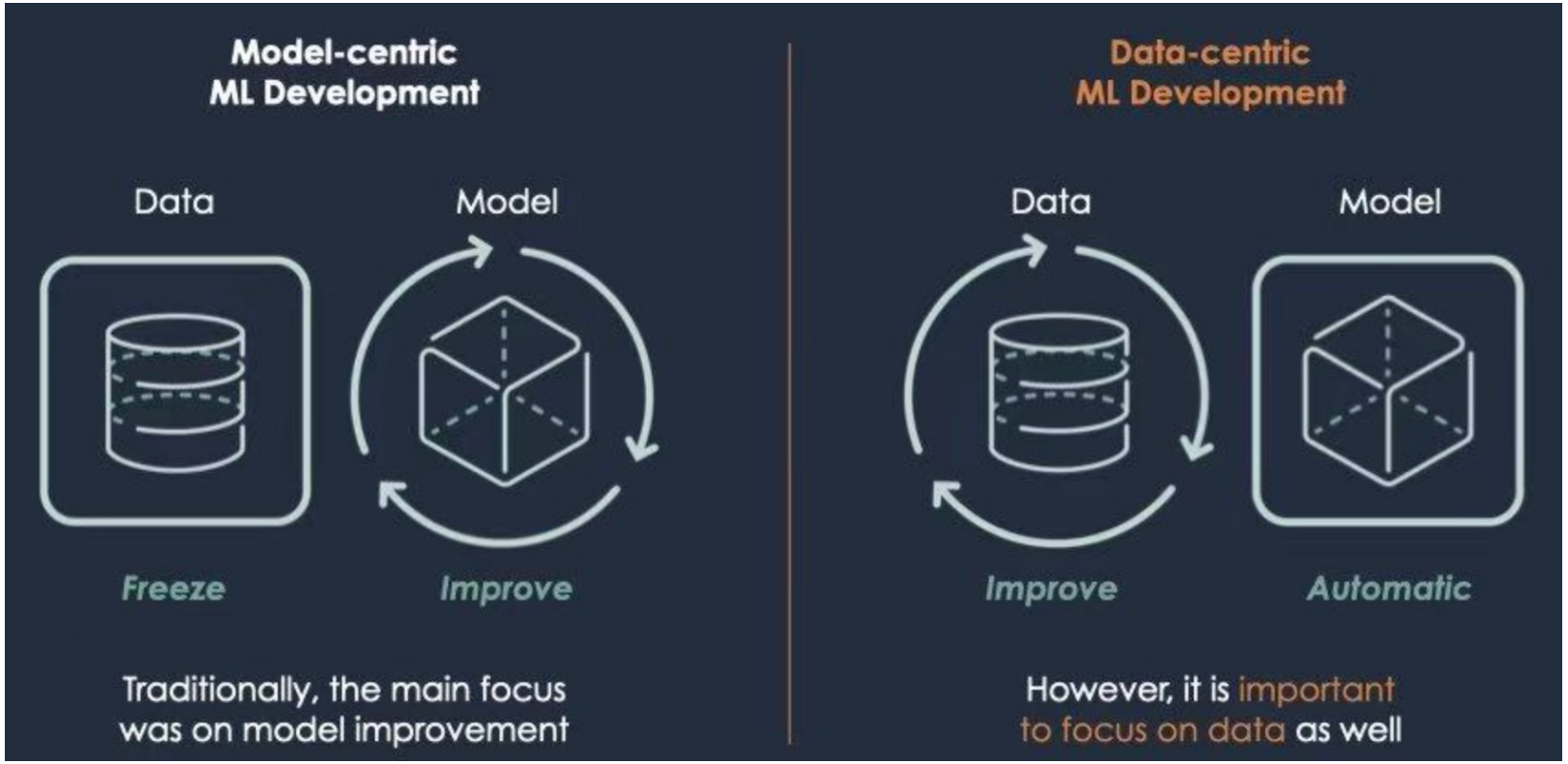
- 2019-2021: Masters of Information and Communication Engineering from Beijing Institute of Technology.

Food and Culture

Weakness Identification of Binary Program Function Using Deep Learning

Security

Computer security, also called cybersecurity, is **the protection of computer systems and information from harm, theft, and unauthorized use**.



Research objectives and analysis

- Objectives:

  - Binary program function weakness identification.

  - Combining related theories of deep learning

- Analysis:

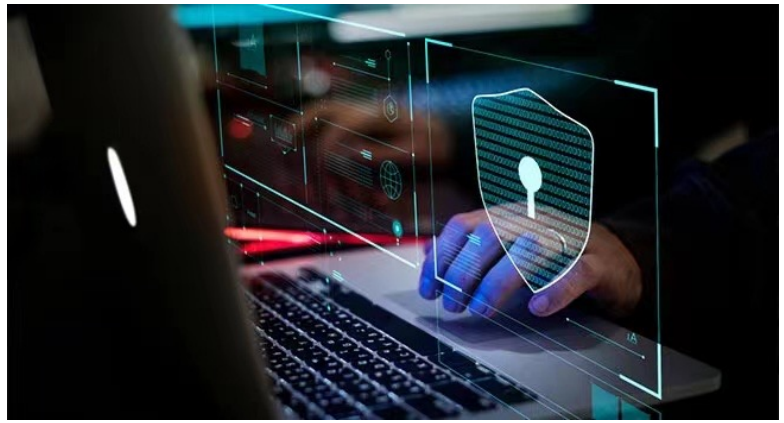
  - Binary program: refers to a program containing binary object code, including executable and linkable files etc.

  - Same-origin vulnerabilities: Refers to binary program vulnerabilities compiled from the same source code

  - Software Vulnerabilities: Refers to errors, weakness or malfunctions in the software that can cause the software to operate in an unexpected manner, resulting in poor or incorrect results

Research background and significance

- In 2022, almost 17,000 vulnerabilities were recorded to Common Vulnerability and Exposures (CVE)
- The research studies the methods of weakness identification using deep learning
- It is of great significance to deal with the increasingly serious threat of software vulnerability
- Detect and evaluate vulnerabilities in binary files to improve systematic detection to increase accuracy and reliability features



Research history

2010

Zimmermann and others used artificial intelligence technology to predict software defects for the first time

2014

Padmanabhuni et al. summarized the vulnerability model and used machine learning and feature engineering to determine whether the x86 binary program has a buffer overflow

2015

Peng et al. proposed a program vector representation generation method based on abstract syntax tree, and introduced deep learning technology to defect discrimination task.

2016

LSTM did consider emerge issues for software programs to identify the vulnerable part of software components and applies further analysis to estimate risk and assign appropriate patches.

2017

Vuddy had a problem of attemption and failure to scale to the size of the OSS code base

2018

Li et al. put forward VulDeePecker, which is the first to use deep neural network to learn source code defect code patterns and realize source code-oriented defect discrimination

2019

Lee et al. proposed Instruction2vec, which divides instructions into elements according to model rules, and combines deep neural networks to implement instruction embedding

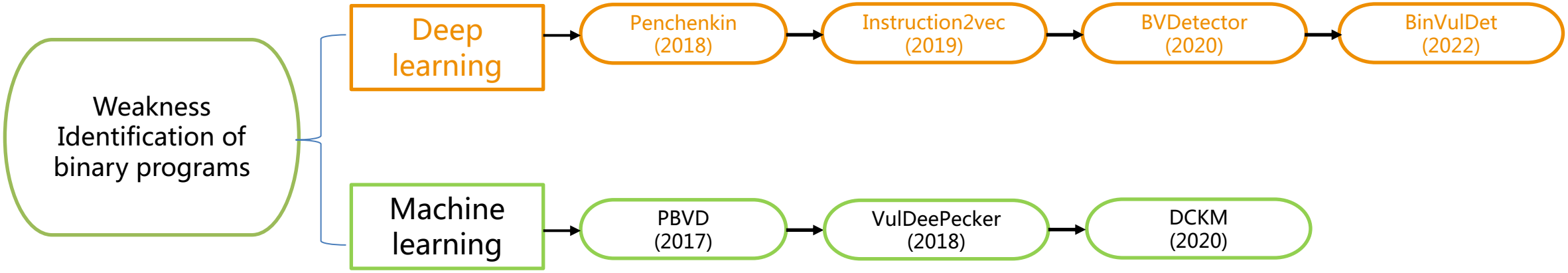
2020

Tian et al. BVDetector, uses slicing technology to construct a more appropriate granularity of defect discrimination input, and combines with deep neural networks to realize binary program-oriented defect discrimination.

2022

Wang et al. proposed a method of decompilation to generate pseudocode enabling accurate identification of known and unknown vulnerabilities in binary code.

Research Trend

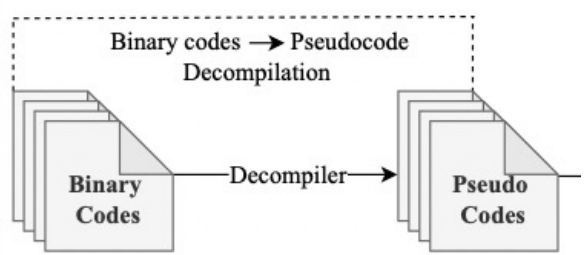


Goals and difficulties

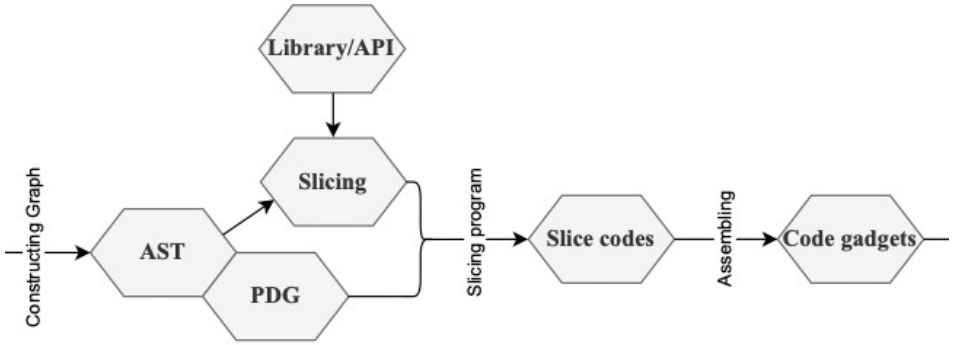
T	Target	Effectively embed and examine binary code to improve systematic detection to increase accuracy and reliability features in binary programs.
I	Input	Juliet Test Suite sample codes & CVE's
P	Process	Using our model to learn the feature of software weakness codes and identify fresh software vulnerabilities
O	Output	The proposed technique tries to identify software vulnerabilities with high efficiency.

P	Problem	Majority of current deep learning-based methods of weakness identification detect vulnerabilities in source code rather than binary code and detect only known vulnerability.
C	Condition	Weakness identification technology is to detect known and unknown vulnerabilities.
D	Difficulty	Building the model and integration of properties.
L	Level	The results is promising and has great optimism.

Framework

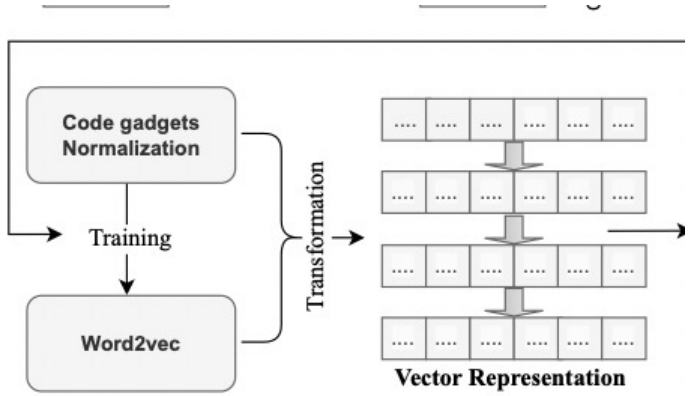


Decompilation procedure to get pseudocode containing high-level semantic features.

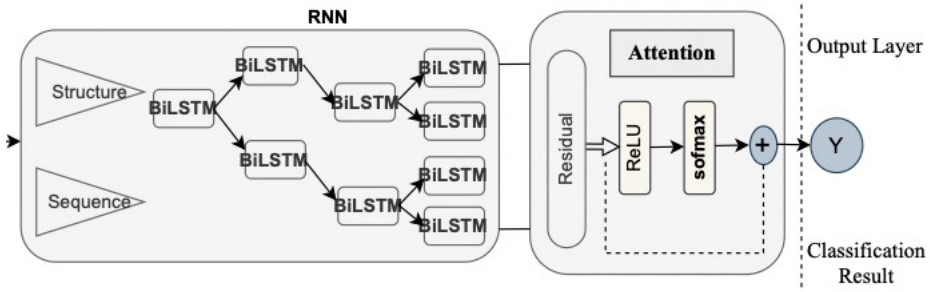


The programme slicing method is then applied to extract the statements containing data dependencies and control dependencies associated to the vulnerability.

Framework



Slice codes are processed using structure and sequence embedding technique with a Tree-based Bi-LSTM



Attention structure is designed with incorporation of Residual connections

Innovation point

- Incorporate of attention mechanisms with residual connections, which are novel techniques that have been shown to improve the performance of neural networks in other domains.

- By integrating these techniques into the proposed model, the performance of the model shall further improve.

Thank you !