UNIVERSITÀ DEGLI STUDI DI UDINE

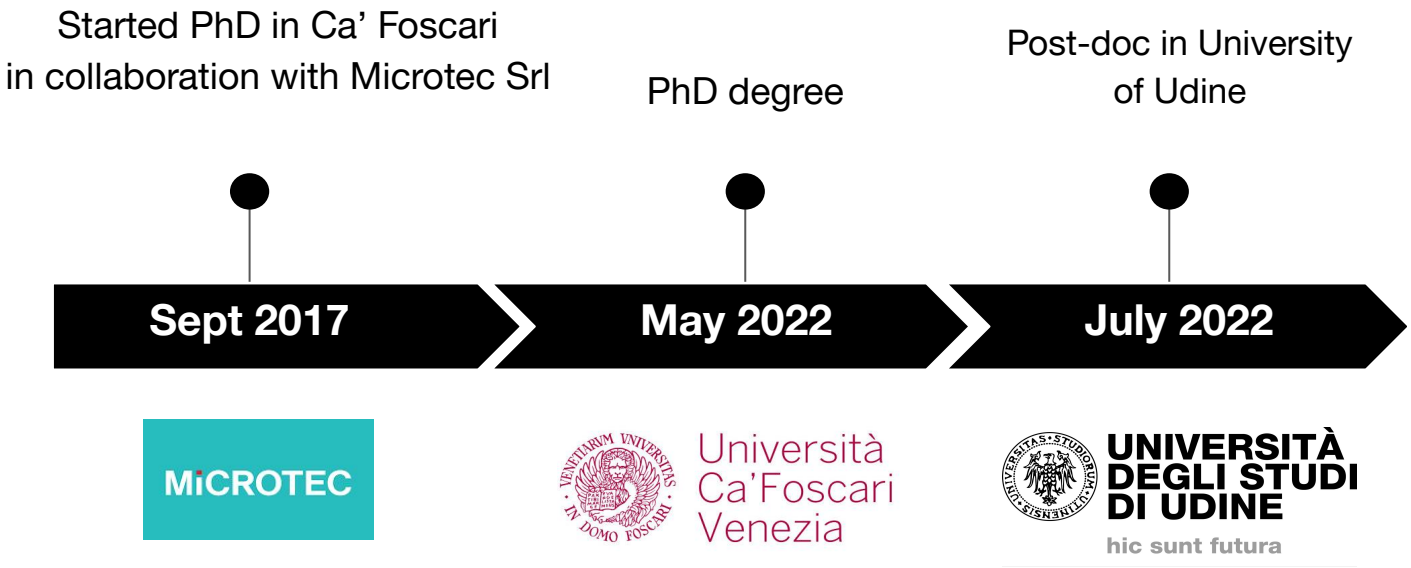
hic sunt futura

PRIN2020 Nirvana Project Update

PhD Dalila Ressi

Supervisors:
Carla Piazza, Sabina Rossi

About Me

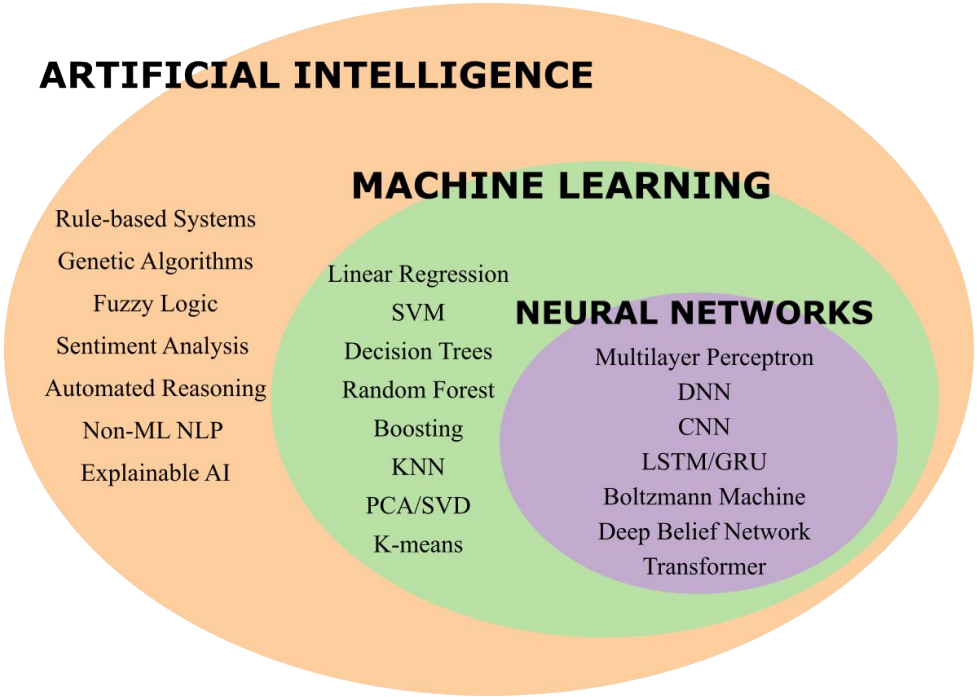


PhD thesis: *"Convolutional Neural Networks Compression for Embedded Industrial Applications"*

Research Works

- “Neural Networks Reduction via Lumping” D.Ressi, R. Romanello, S. Rossi, C. Piazza (Aixia 2022, working on journal edition)
- “AI-Enhanced Blockchain Technology: a Review of Advancements and Opportunities” D.Ressi, R. Romanello, S. Rossi, C. Piazza (Submitted to JNCA)
- “Security Verification of Ethereum Smart Contracts with ML Taking a Free Ride on Static Analysis” D.Ressi, R. Romanello, S. Rossi, C. Piazza, M. Bugliesi, S. Crafa (DLT2023, oral communication)

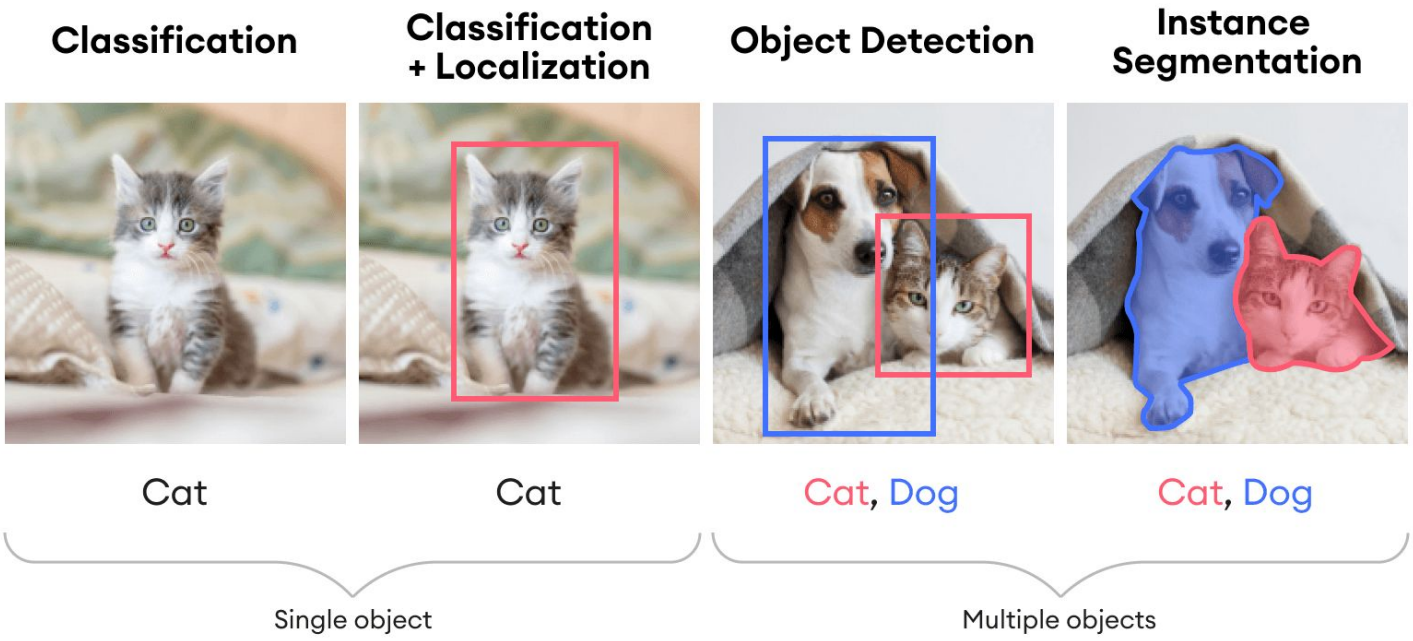
Artificial Intelligence Background



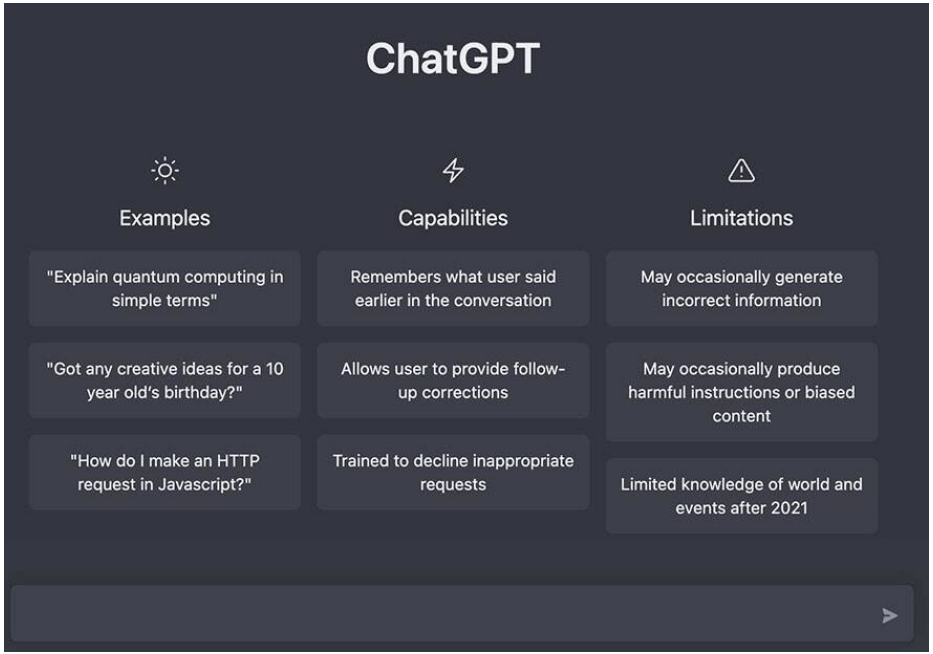
Tasks that can be solved with AI are mainly divided into three categories:

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

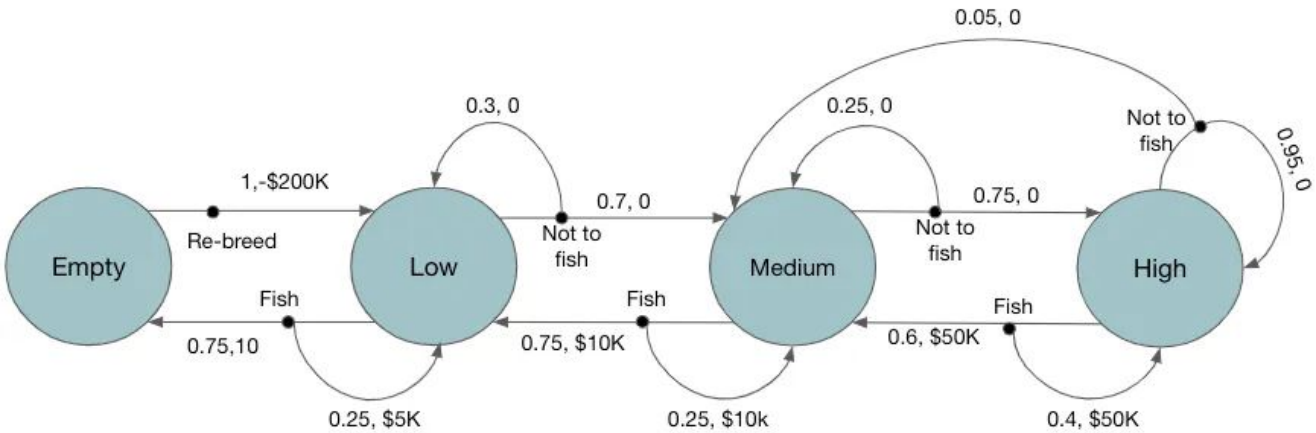
Supervised Learning: Image Recognition



Unsupervised Learning: AI-generated content



Reinforcement Learning: Markov Decision Process



Transition graph of fishing salmons markov decision process:
RL can be used to decide if to fish or not to fish to maximize reward over a certain period of time

AI-Enhanced Blockchain Technology: a Review

During this year we mainly worked on a survey about AI/ML and blockchain (BC)

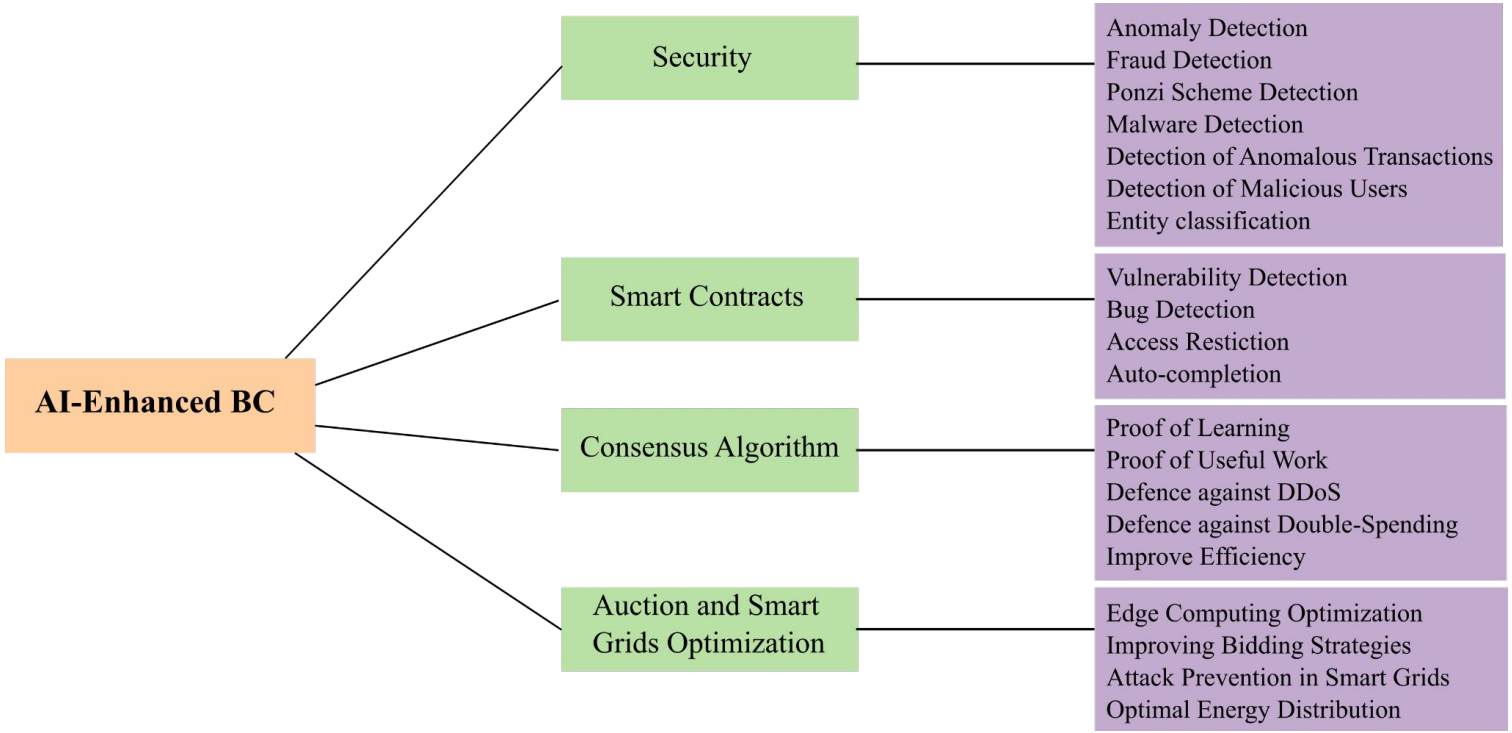
BC + AI in many application areas:

- Healthcare
- IoT/IoV/Smart Cities
- DeFi
- Cryptocurrency

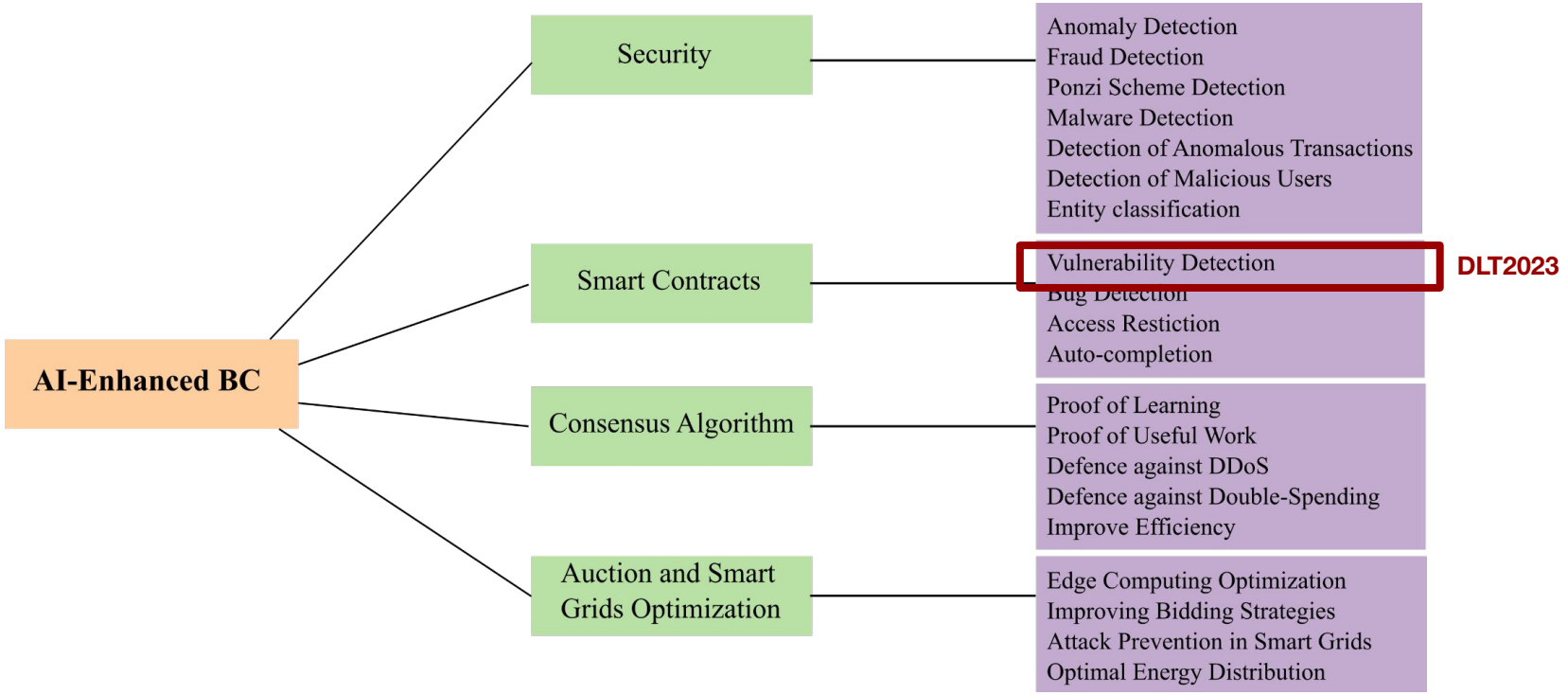
Areas improved through AI:

- Security
- Consensus Algorithm
- Auction and Smart Grids Optimization
- Smart Contracts

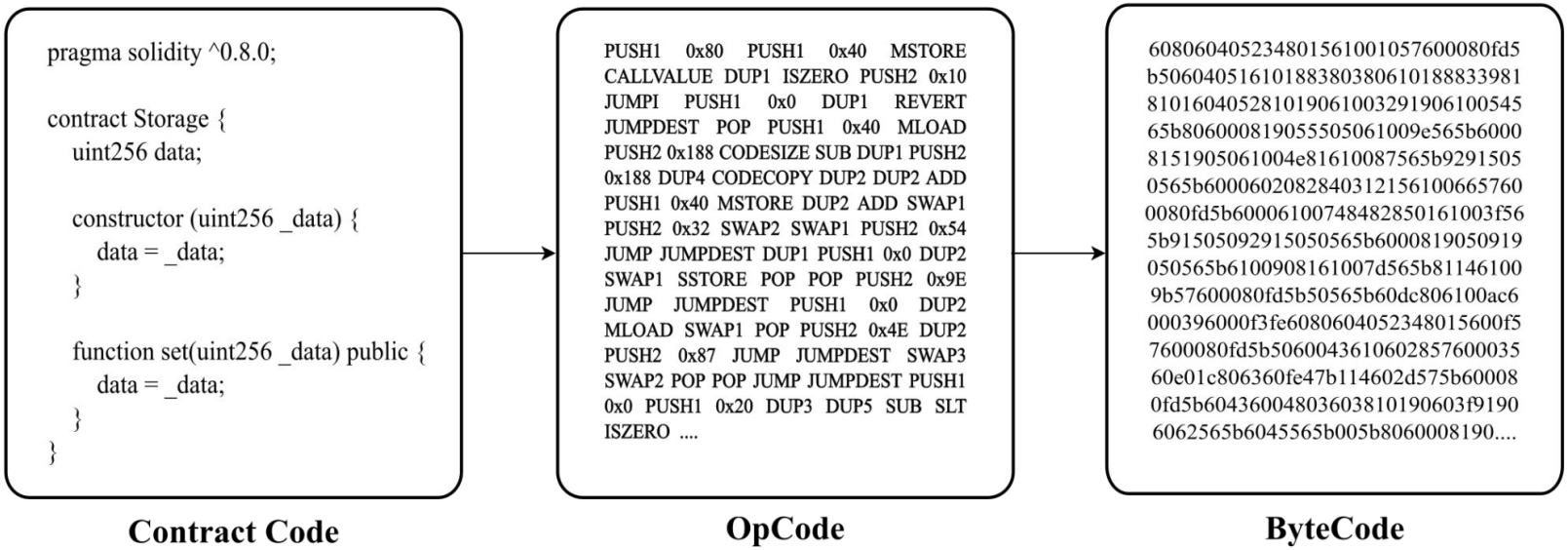
Areas Currently Improved Through to AI/ML



Areas Currently Improved Through to AI/ML



Ethereum Smart Contracts



Vulnerabilities in Ethereum Smart Contracts

Denial of Service
“tx.origin” usage
Integer Overflow/Underflow
Re-entrancy
Call to the unknown
Gassless “send”
…

→ Solidity Programming Language

Immutable bugs/mistakes
Ether lost in transfer
…

→ Ethereum Virtual Machine

Timestamp dependency
Transaction Ordering Dependency
…

→ Ethereum Blockchain Design

Vulnerabilities in Ethereum Smart Contracts

Denial of Service
“tx.origin” usage
Integer Overflow/Underflow
Re-entrancy
Call to the unknown
Gassless “send”
…

→ Solidity Programming Language

Immutable bugs/mistakes
Ether lost in transfer
…

→ Ethereum Virtual Machine

Timestamp dependency
Transaction Ordering Dependency
…

→ Ethereum Blockchain Design

⇨ More than 20 types of different vulnerabilities, we focus only on detection

Vulnerability Detectors: Formal Verification Techniques

Most of existing frameworks exploit static analysis, some examples are:

- Slither
- Mythril
- Oyente
- Securify
- SmartCheck
- SmartScan
- …

Usually specialized on specific security properties,
running multiple frameworks is computationally expensive

Static analyzers can be divided into exact methods (‘hardcore’)
and approximated techniques

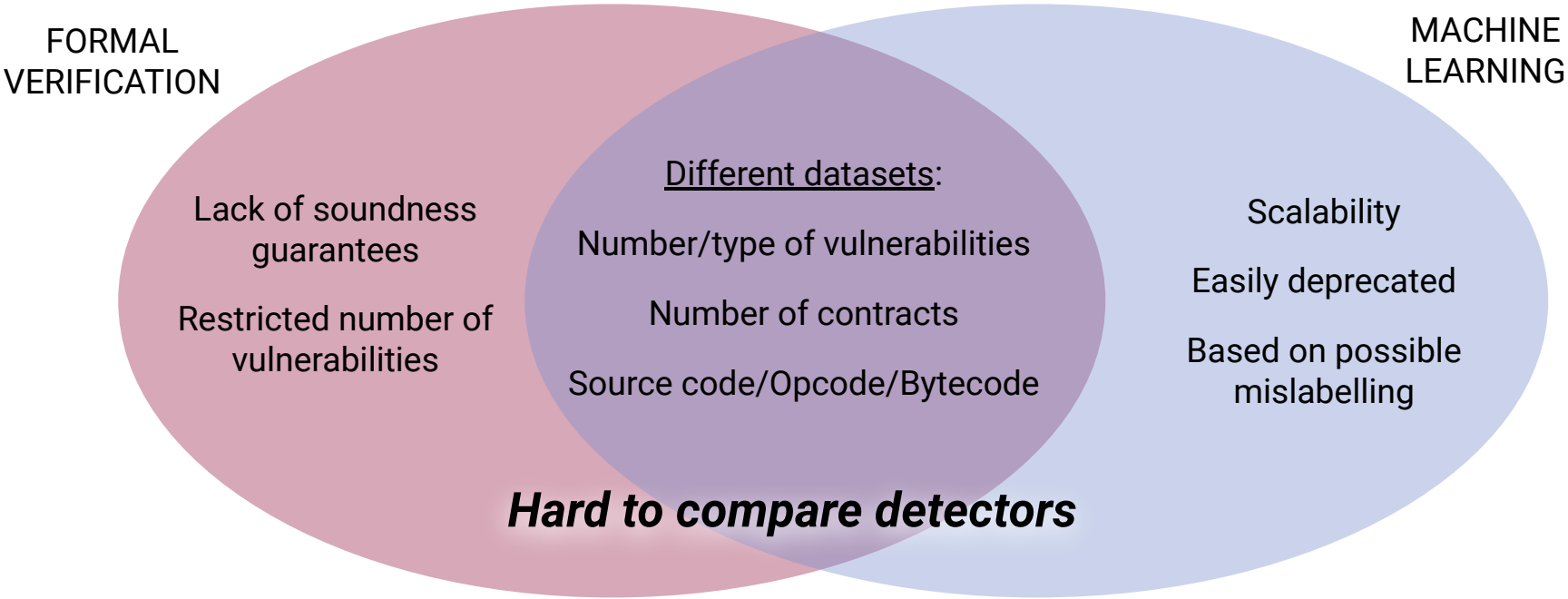
Vulnerability Detectors: ML Techniques

ML learning techniques take advantage of existing frameworks for dataset labeling, and then they simply perform classification with techniques such as:

- SVM
- **Boosting**
- Random Forest
- Decision Tree
- CNN
- GNN
- **LSTM**
- …

Machine Learning techniques claim to have higher performance than static analyzers with respect to inference time and accuracy

Limitations and Open Problems



Future Work: Dataset Creation

The availability of **benchmark dataset** would not only provide a solid base to develop new detection algorithms, but it would also allow us to evaluate and compare existing ones.

Key features:

- Large number of smart contracts
- Include all three representations of a contract (source code, opcode, bytecode)
- Metadata

Labeling process:

- Exploit only formal verification techniques providing soundness guarantees
- Represent as many as possible vulnerabilities (eventually augmenting the cardinality of heavily underrepresented classes)

Thank You!

dalila.ressi@uniud.it