UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II
Scuola Politecnica e delle Scienze di Base
Corso di Laurea Magistrale in Ingegneria Informatica

Tesi di Laurea Magistrale in Software Security

Una strategia per Ottimizzare l'Addestramento degli IDS attraverso la Data Augmentation

Anno Accademico 2023/2024

relatore
Ch.mo Prof. Roberto Natella

correlatore
Ing. Simona De Vivo

candidato
Luigi Cerrato
Matr. M63001402

Il contesto

- Lo sviluppo delle infrastrutture IoT, oltre a portare molti vantaggi, comporta **nuove sfide** per la **sicurezza informatica.** Questi contesti, infatti, sono spesso bersaglio di diversi attacchi tra cui i **Distributed Denial of Service** (**DDoS**)

Assumono un ruolo fondamentale gli **Intrusion Detection Systems** (**IDS**).

- *Classificazione degli attacchi*
- *Anomaly / Signature Based*
- *Posizionabili in vari punti dell'infrastruttura*

Le sfide nell'addestramento

- Mancanza di **dataset** bilanciati e realistici
- Difficoltà nel riconoscere nuovi tipi di attacchi (zero-day)
- Necessità di **bilanciare** falsi positivi e falsi negativi

Soluzione Proposta: Utilizzare la tecnica di **Data Augmentation** con il supporto del simulatore **DDoShield-IoT** per arricchire i dataset e migliorare le prestazioni.

DDoShield-IoT

- FTP, RTMP, HTTP e Mirai Botnet per il traffico malevolo

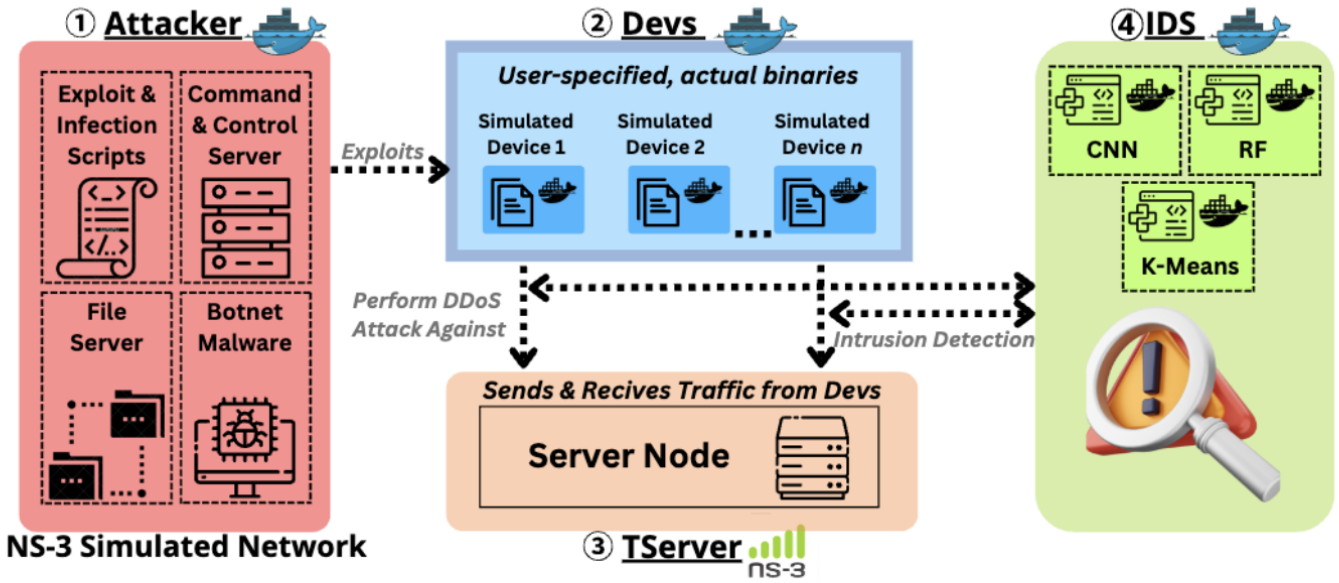


Fig. 1. DDoSHIELD-IoT Overview

TON_IoT Dataset

- **Telemetry:** Dati estratti dai sensori IoT
- **Operating Systems:** Sistemi Operativi Windows e Linux
- **Network:** Dati estratti dalla rete
- Traffico **Benigno** (FTP, HTTP, MQTT...)
- Traffico **Maligno** generato da Offensive Systems(Kali Linux)
  - Nove **Famiglie** (DDoS, XSS, Injection, MITM, backdoor…)

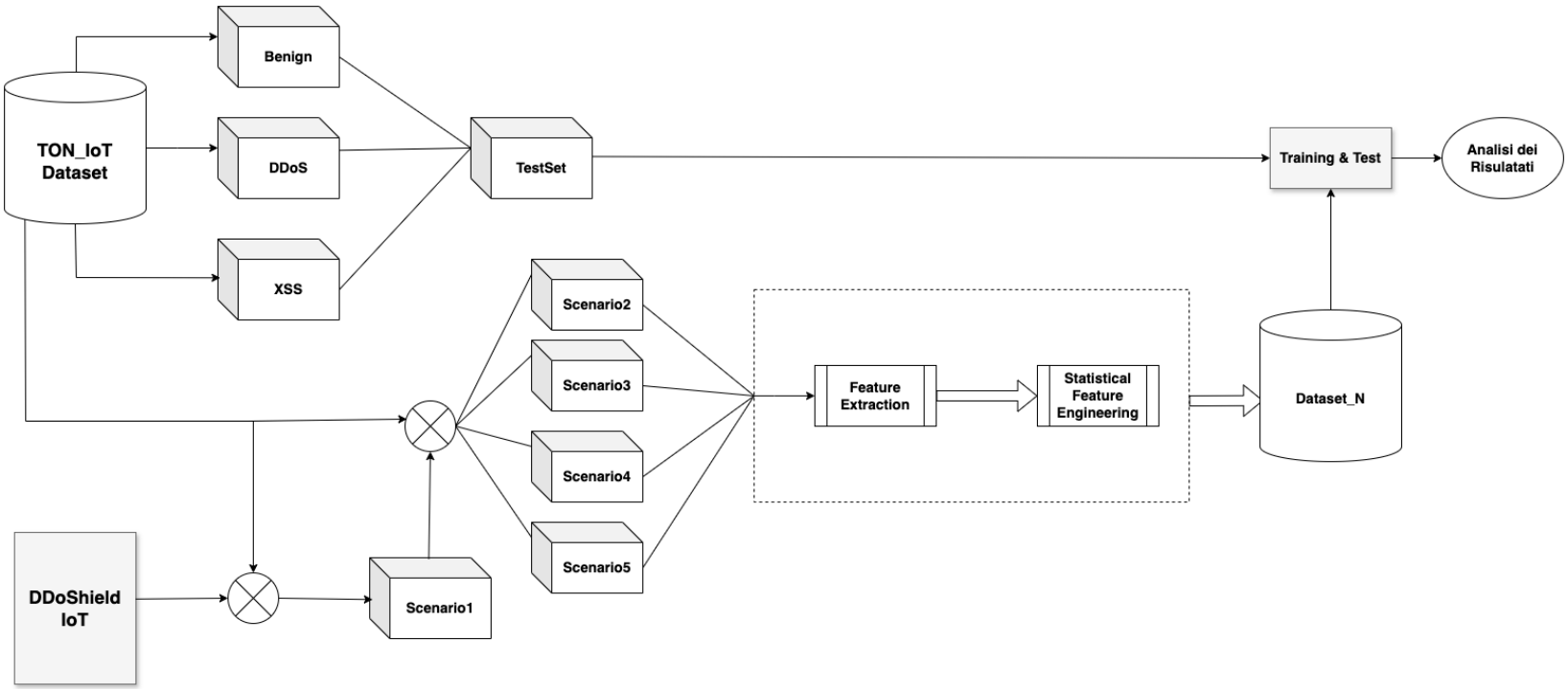
Data Augmentation

- Tecnica usata per **aumentare** la quantità dei dati di adestramento
- Utilizzando il simulatore DDoShield possiamo generare **nuovo traffico**
- Particolarmente utile con dataset di **pochi dati** o dataset **sbilanciati**

Algoritmi – Anomaly Based

- Un IDS con approccio Anomaly Based identifica attività che si discostano da un modello di comportamento "normale" predefinito.
- **K-Means:** Un algoritmo di clustering non supervisionato che raggruppa I dati in **cluster**, identificando le similarità tra le istanze.
- **Random Forest (RF):** Algoritmo di apprendimento supervisionato basato su un insieme di alberi decisionali, noto per la sua robustezza e capacità di gestire dati complessi e rumorosi.
- **Convulational Neural Network (CNN):** Reti **neurali** profonde progettate per l'analisi di dati strutturati, come immagini e sequenze temporali

Workflow


Benign
TON_IoT Dataset
DDoS
XSS
TestSet
Training & Test
Analisi dei Risulatati
Scenario2
Scenario3
Scenario4
Scenario5
Feature Extraction
Statistical Feature Engineering
Dataset_N
DDoShield IoT
Scenario1

Workflow

1. Definizione dei diversi scenari: Identificazione dei dataset da generare e delle proporzioni di traffico tra le sorgenti Ton_IoT e DDoShield.

2. Raccolta dei dati: Creazione dei file .pcap per ogni scenario.

3. Estrazione delle feature d'interesse: Analisi dei dataset al fine di estrarre le feature rilevanti e migliorare la qualità dei dati.

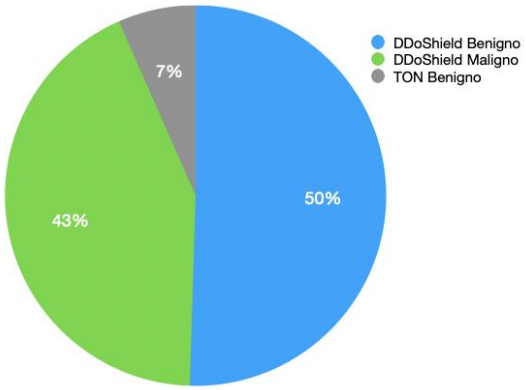
4. Creazione di un Test-Set: Generazione un set di dati di test distinto da quelli utilizzati per l'addestramento (dati da Ton_IoT).

5. Addestramento dei modelli: Fase di train, test e misurazione delle performance degli algoritmi di apprendimento.

6. Analisi e discussione dei Risultati: Valutazione delle metriche e conclusioni.

Scenario 1

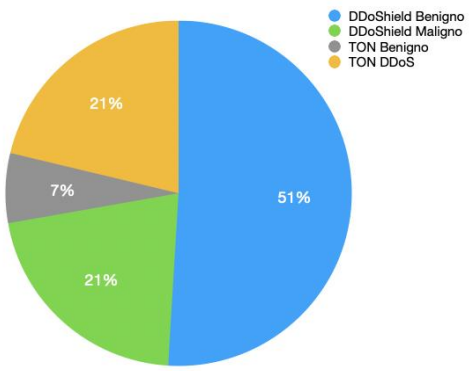
- Dataset di base, bilanciato.
- Frazioniamo ora il traffico maligno generando nuovi scenari DDoS



	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
K-means	84.09	76.54	98.95	86.31
RF (0.25)	70.74	100	42.17	59.32
CNN	66.34	99.66	33.42	50.06

Scenario 2

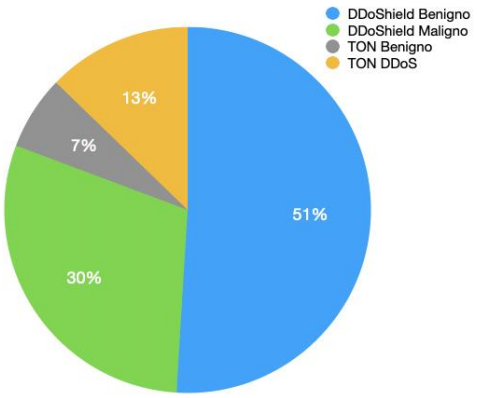
- Dataset 50% simulatore – 50% Ton



	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
K-means	86.53	79.60	98.75	88.14
RF (0.25)	97.40	95.11	100	97.49
RF (0.50)	99.95	99.90	100	99.95
CNN	99.98	100	99.97	99.98

Scenario 3

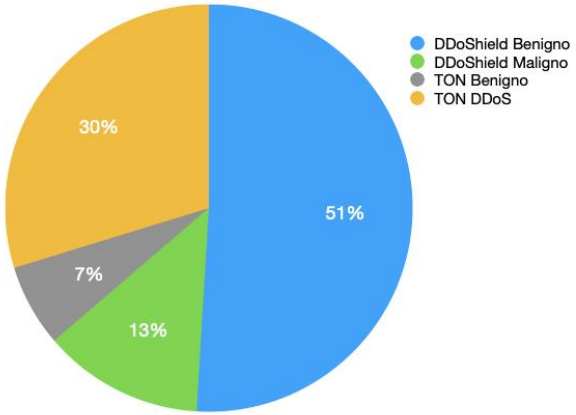
- Dataset 70% simulatore – 30% Ton



	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
K-means	86.62	80.71	96.74	88.00
RF (0.25)	95.11	91.18	100	95.38
RF (0.50)	100	100	100	100
CNN	67.90	100	36.41	53.38

Scenario 4

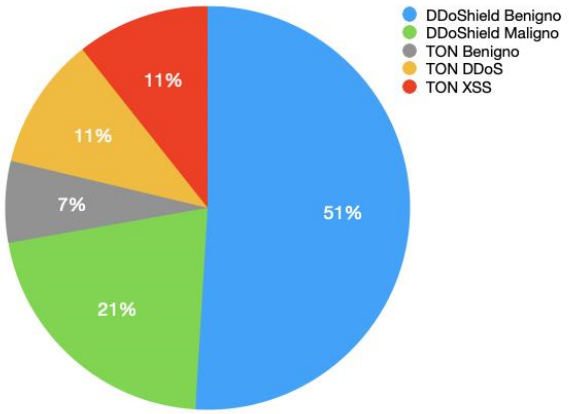
- Dataset 30% simulatore – 70% Ton



	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
K-means	84.07	78.26	94.97	85.81
RF (0.25)	93.30	88.28	100	93.78
RF (0.50)	99.91	99.83	100	99.91
CNN	99.60	100	99.22	99.60

Scenario 5

- Dataset 50% simulatore – 50% Ton (DDoS + XSS)



	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
K-means	86.31	86.42	94.48	90.27
RF (0.25)	76.48	74.04	100	85.08
RF (0.50)	99.99	99.99	100	99.99
CNN	96.96	99.96	95.50	97.68

Considerazioni

- **K-Means**: Bilanciato in performance e «comportamento», lavora bene con i falsi negativi.

- **Random Forest** (**RF**): Ottime prestazioni. Coglie relazioni non lineari e complesse → lavora bene anche con dataset eterogenei ed in condizioni in cui non si hanno molti dati e/o sbilanciamenti.

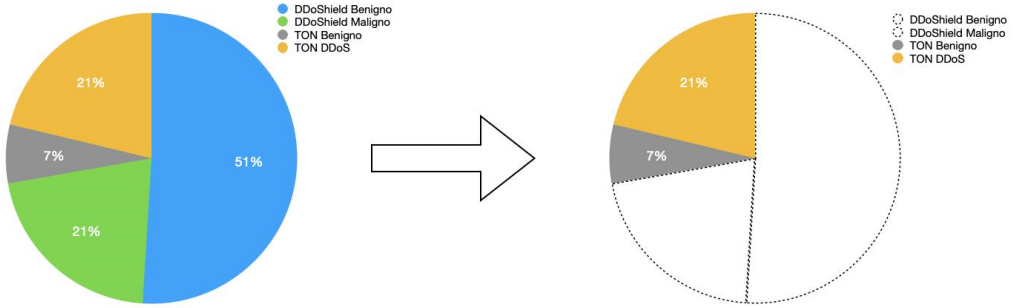
- **Convolutional Neural Network** (**CNN**): Difficoltà con sbilanciamenti e risulta conservativo in alcuni casi (Dataset 1 e 3).
  Necessità di una quantità maggiore di dati → basse performance in alcuni scenari con pochi dati.

L'impatto della Tecnica

- Cosa succede se rimuovo il contributo del simulatore?
- In parentesi [ ] sarà riportato il risultato utilizzando il simulatore.

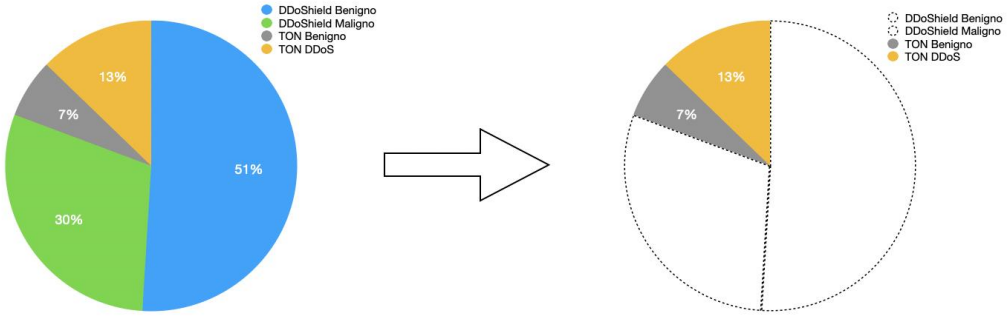
Obiettivo : Valutare l'impatto della tecnica, individuando il contesto operativo più favorevole.

Dataset2 Ton_Only



	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
K-means	98.47 [86.53]	97.49 [79.60]	99.57 [98.75]	98.52 [88.14]
RF (0.25)	49.34[97.40]	100 [95.11]	0.02 [100]	0.05 [97.49]
RF (0.50)	49.34 [99.95]	0 [99.90]	0 [100]	0 [99.95]
CNN	93.16 [99.98]	99.68 [100]	86.77 [99.97]	92.78 [99.98]

Dataset3 Ton_Only

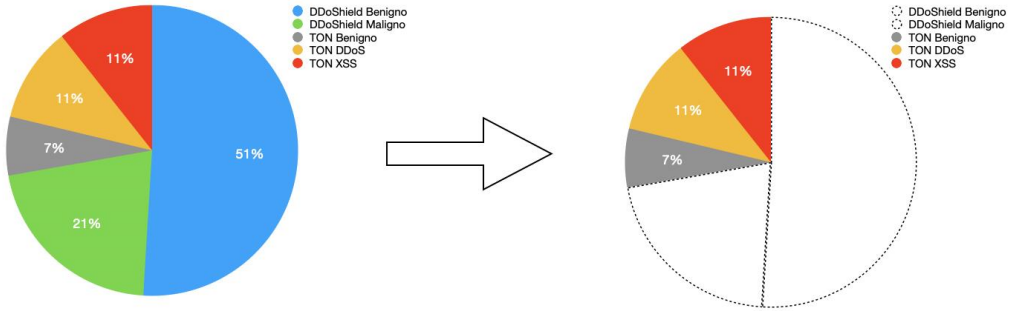


	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
K-means	97.40 [86.62]	95.15 [80.71]	100 [96.74]	97.51 [88.00]
RF (0.25)	57.72 [95.11]	54.51 [91.18]	100 [100]	70.56 [95.38]
RF (0.50)	86.44 [100]	78.12 [100]	100 [100]	88.29 [100]
CNN	98.89 [67.90]	97.87 [100]	100 [36.41]	98.92 [53.38]

Dataset4 Ton_Only

	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
K-means	97.03 [84.07]	94.50 [78.26]	100 [94.97]	97.17 [85.81]
RF (0.25)	57.18 [93.30]	54.14 [88.28]	100 [100]	70.00 [93.78]
RF (0.50)	86.21 [99.91]	78.46 [99.83]	100 [100]	88.46 [99.91]
CNN	98.82 [99.60]	97.72 [100]	100 [99.22]	98.85 [99.60]

Dataset5 Ton_Only



	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
K-means	97.85 [86.31]	96.03 [86.42]	99.92 [94.48]	97.94 [90.27]
RF (0.25)	52.28 [76.48]	51.50 [74.04]	100 [100]	67.99 [85.08]
RF (0.50)	99.90 [99.99]	99.92 [99.99]	100 [100]	99.91 [99.99]
CNN	98.50 [96.96]	98.25 [99.96]	99.85 [95.50]	98.92 [97.68]

Considerazioni

- **K-Means**: Introdurre il simulatore, riduce l'overfitting complicando le predizioni
- **RF**: Ottime Prestazioni, algoritmo robusto che cattura relazioni complesse → beneficia di una generalizzazione e cattura nuovi pattern di attacco.
- **CNN**: Le reti neurali dipendono molto di più della distribuzione e complessità dei dati di addestramento. In alcuni casi il simulatore potrebbe fornire dati aggiuntivi, in altri casi potrebbe essere utile per generare traffico e ridurre l'overfitting (Dataset 3).
- Con questa tipologia di algoritmi è necessario non sbilanciare troppo il dataset ed individuare il giusto trade-off.

Grazie per l'attenzione !