REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEURE ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE IBN KHALDOUN - TIARET

MEMOIRE

Présenté à :

FACULTÉ DES MATHEMATIQUES ET DE l'INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité : Réseau et Télécommunication

Par :

GACEM Fatima Zahraa
FERNANE Souad

Sur le thème

Vers une approche pour les systèmes de détection d'attaque de réseau basée sur données structurées de graphe

---

Soutenu publiquement le 29 / 06 / 2022 à Tiaret devant le jury composé de :

Mr BENDAOUD Mebarek	Pr	Université Tiaret	Président
Mr DAOUD Mohamed Amine	MAA	Université Tiaret	Encadrant
Mr ALEM Abdelkader	MAA	Université Tiaret	Examinateur

2021-2022

Remerciement

*En premier lieu, nous remercions **Dieu « ALLAH »** le très haut qui nous a donné le courage et la volonté d'accomplir ce modeste travail.*

*En seconde lieu Mr : **DAOUD Mohamed Amine** notre encadreur, il s'est toujours montré à l'écoute et très disponible tout au long de sa réalisation et auquel nous tenons à témoigner nos reconnaissances et notre gratitude les plus sincères.*

Nos vifs remerciements vont aux membres du jury d'avoir accepté d'examiner notre travail, à savoir :

*Monsieur **BENDAOUD Mebarek** et monsieur **ALEM Abdelkader** de notre université.*

Nous tenons à saisir cette occasion et adresser nos profonds remerciements et reconnaissances à toutes personnes qui nous ont aidés de près ou de loin dans la réalisation de ce mémoire.

Enfin nous exprimons notre profonde reconnaissance à tous responsables et enseignants de l'université de Tiaret qui ont contribué à notre formation.

Dédicaces

Je dédie ce mémoire à :

Mes parents :

Ma mère, qui a œuvré pour ma réussite, par son amour, son soutien et les sacrifices consentis, pour toute son assistance et sa présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.

Mon père, qui peut être fier et trouver ici le résultat de longues années de sacrifices et de privations pour m'aider à avancer dans la vie. Puisse ALLAH faire en sorte que ce travail porte son fruit ; merci pour l'éducation et le soutien permanent venu de toi.

Mes sœurs et mon frère qui n'ont cessé d'être pour moi des exemples de persévérance, de courage et de générosité.

Mon grand-père qui nous a quittés.

Mes tantes et mes oncles, ainsi qu'à toute ma famille.

Ma chère collègue Fatima avec qui je partage ce travail qui n'a jamais cessée de me soutenir, pour son entente et sa sympathie

Mes professeurs qui doivent voir dans ce travail la fierté d'un savoir acquis.

Souad

Dédicaces

Avec joie, fierté et respect, Je dédie ce mémoire :

À ma très chère mère ***Fatiha***

Et mon très cher père ***Kadda***

Pour tous leurs sacrifices, leur amour, leur tendresse, leur soutien et prière tout au long de mes études.

À ma chère grand-mère ***Aicha***

À mes chers frères ***Mohamed Yacine*** *et* ***Abdelkader***

À mes chères sœurs ***Amina, Aouli*** *et* ***Asmaa*** *et leurs maris ;*

Pour leur appui et leur encouragement

À mes nièces et neveux chacun son nom.

À mon cher binôme ***Souad***

Pour son soutien moral, sa patience et sa compréhension tout au long de ce projet. Que Dieu nous garde toujours ***unis***

Enfin je le dédie à tous mes amis que je n'ai pas cités et à tous ceux qui me connaissent.

Fatima

Résumé

Les attaques informatiques sont contrôlées et affectée par plusieurs facteurs, tels que l'environnement, le mode de détection etc., il est difficile de détecter ces attaques avec précision, ce qui impact sur le déroulement des réseaux malgré la proposition des solutions IDS dont certaines ont des avancées particulières. La situation rend l'amélioration des systèmes de détection d'intrusion est indispensable via les technologies informatiques telles les techniques de machine Learning et les graphes de connaissances.

Dans ce travail, Une modeste idée a été proposée afin d'extraire et de classifier des nouvelles connaissances pour les IDS réseau par la combinaison des techniques de machine Learning et les graphes de connaissances. Cette expérience, nous permet à découvrir les relations entre données du data-set CIDDS-001 utilisé, cela fournit intuitivement une interprétation raisonnable des résultats descriptives.

Mots clés : IDS, graphe de connaissance, Machine Learning, classification.

ملخص

يتم التحكم في هجمات الكمبيوتر التي تتأثر بعدة عوامل، مثل البيئة وطريقة الكشف وما إلى ذلك و من الصعب اكتشاف هذه الهجمات بدقة مما يؤثر على عمل الشبكات على الرغم من اقتراح حلول IDS، التي لها تطورات معينة. هذا الموقف يجعل تحسين أنظمة اكتشاف التسلل أمرًا ضروريًا بواسطة تقنيات الكمبيوتر مثل تقنيات التعلم الآلي والرسوم البيانية المعرفية.

في هذا العمل، تم اقتراح فكرة متواضعة لاستخراج وتصنيف المعرفة الجديدة ل IDS الشبكات من خلال الجمع بين تقنيات التعلم الآلي والرسوم البيانية المعرفية. تسمح لنا هذه التجربة باكتشاف العلاقات بين معلومات مجموعة البيانات CIDDS-001 المستخدمة، مما ينتج تفسيرًا معقولًا للنتائج الوصفية.

الكلمات المفتاحية:IDS ، الرسم البياني المعرفي، التعلم الآلي، التصنيف.

Abstract

Computer attacks are controlled and affected by several factors, such as the environment, the mode of detection etc., it is difficult to detect these attacks with precision, which has an impact on the operation of networks despite the proposal of IDS solutions, some of which have particular advances. The situation makes the improvement of intrusion detection systems essential via computer technologies such as machine learning techniques and knowledge graphs.

In this work, a modest idea has been proposed to extract and classify new knowledge for network IDS by the combination of machine learning techniques and knowledge graphs. This experience allows us to discover the relationships between data of the CIDDS-001 data-set used, this intuitively provides a reasonable interpretation of the descriptive results.

Keywords: IDS, knowledge graph, Machine Learning, classification.

Table des matières

La liste des figures :

Liste des tableaux

Acronyms

IDS: Intrusion Detection System

HIDS: Host based Intrusion Detection System

NIDS : Les systèmes de détection d'intrusions réseau

NLP: Natural Language Processing

RDF: Resource Description Framework

KG: knowledge Graph

IA: Intelligence artificielle

ML: Machine Learning

BD : Base de Données

K-NN : K Nearst Neighbors

ACID : Atomicité, Cohérence, Isolation, Durabilité

CIDDS : Coburg Intrusion Detection Data Sets

Introduction générale :

Avec le développement de nouvelles technologies et applications de l'information, l'échelle du cyberespace s'étend progressivement de l'Internet traditionnel à une variété de domaines tels que la fabrication, la santé, l'agriculture, l'aviation, les affaires, etc. En conséquence, il peut comprendre des interactions entre les systèmes, ce qui le rend une infrastructure plus complexe. A cet effet, des opportunités d'attaques se multiplient, en raison de la combinaison de cyber et de nombreux physiques actifs, les conséquences des cyberattaques deviennent de plus en plus graves. Compte tenu du nombre et de l'intensité croissante des attaques, le développement des outils de sécurité est une tache cruciale.

La persistance des attaques modernes fait à des limites des technologies de défense traditionnelles basées sur des règles d'experts, l'apprentissage automatique et l'apprentissage en profondeur sont devenues de plus en plus apparentes. Les tâches relativement simples, telles que l'extraction de caractéristiques, la détection d'anomalies et la classification des données ne peuvent plus restaurer l'image complète du comportement d'attaque. Les connaissances d'experts cachées dans les données de sécurité constituent toujours une percée très importante pour résoudre les problèmes. Cependant, les données liées aux systèmes de détection d'intrusion (IDS) ont connu une croissance explosive. Elles sont diverses, hétérogènes et fragmentées, ce qui rend difficile pour les responsables de la sécurité de trouver rapidement les informations dont ils ont besoin. Par conséquent, le problème actuel de l'analyse de la sécurité n'est pas le manque d'informations disponibles, mais comment assembler des informations provenant de plusieurs sources, pour mieux comprendre la situation et fournir une aide à la décision.

La majorité des solutions des systèmes de détection d'intrusion (IDS) ont été utilisée par l'intégration des techniques de machine Learning (ML) afin de classifier les attaques qui sont liés par leur appartenance à la même classe, mais cette application de ML, elle est appuyée sur la compréhension de connaissances à l'aide de données en colonnes, dont les lignes de données sont traites indépendamment des autres. Ils existent un nombre de facteurs sont intuitivement accepter par les spécialistes du domaine de la détection des cyber-attaques, mais il peut y avoir d'autre facteurs, y compris ceux qui n'ont pas encore été découverts et ne peuvent pas être expliques par l'utilisation des techniques de machine Learning. Cette relation n'existe pas avant la classification et peut maintenant être utilisée pour créer les graphes de connaissance (Knowledge Graphs).

Les graphes de connaissances (KG Knowledge graphe) constituent un réseau sémantique composé des nœuds et des arcs, fournissant une méthode de modélisation intuitive pour divers scénarios d'attaques et de défense. Les nœuds peuvent être des entités ou des concepts abstraits les arêtes représentent les attributs ou les relations entre les entités. Les relations entre les points de données sont précalculées et deviennent une partie importante de l'ensemble de données. Cela signifie que non seulement chaque point de données peut être analysé rapidement et à grande échelle, mais également chaque relation. Leurs avantages peuvent être discutés sous trois aspects : premièrement, en utilisant des techniques de construction et de raffinement du KG, y compris l'ontologie, l'extraction d'informations (IE) et

la désambiguïsation des entités. Deuxièmement, il peut exprimer les connaissances dans le domaine des IDS de manière structurelle et relationnelle, et visualiser les connaissances de manière graphique, ce qui est très intuitif et efficace. Troisièmement, en utilisant des technologies de modélisation sémantique, d'interrogation et de raisonnement. Les graphes de connaissances résolvent le problème du filtrage manuel des informations inutiles, ce qui a une signification pratique pour une détection intelligente.

Pour cela la question de recherche que nous devons poser pour notre travail est : QR : *Est ce qu'il est utile d'extraire des informations d'IDS et de classifier les attaques par la combinaison des KG et ML ?*

L'objectif de ce mémoire est de motiver et de donner une introduction à l'application et la combinaison les techniques ML et KG, cela fait par la création d'une approche en utilisant les données orientes graphe et les techniques du ML qui reprennent la théorie des graphes pour représenter et stocker les données. Ce qui rendent leurs traitements très efficaces pour traiter les relations entre les données du DataSet. Cette proposition résolve le problème du filtrage manuel des informations inutiles, ce qui a une signification pratique pour une détection intelligente.

Les principales contributions de ce modeste travail :

- Représentation facile des données, ce qui fournit un moyen très simple de représenter des relations et des données.
- Exécution plus rapide par la facilite de récupérer et de parcourir plus de données connectées.

Le plan de ce projet de fin d'étude s'articule autour de quatre chapitres :

Chapitre 1 : présente les systèmes de détection d'intrusions, ces types, ces différentes architectures et ces méthodes de détection utilisées.

Chapitre 2 : expose les notions fondamentales des graphes de connaissance ainsi que les techniques adoptées pour sa création, son architecture, les modèles de KG et ces applications.

Chapitre 3 : concerne les techniques de machine Learning qui sont une étape essentielle pour la construction de notre approche.

Chapitre 4 : présente l'implémentation des deux algorithmes de classification supervisée sur la bases de données « CIDDS-001 » par l'outils « NEO4J ».

En dernier lieu, une conclusion générale et des perspectives de ce travail seront présentées.

Chapitre1

Les systèmes de détection d’intrusion

Introduction

Les systèmes et réseaux informatique contient diverses formes de vulnérabilité. Pour faire face à ses problèmes de sécurité, différents mécanismes ont été mis en place pour prévenir toutes sortes d'attaque comme les pares-feux, l'authentification et les proxy…etc.

Malheureusement ces mécanismes ont des limites où certains types des attaques que voulait contourner pour nuire la confidentialité, l'intégrité ou la disponibilité. Dans ce sens la question qui se pose à tous les niveaux et tend à devenir un enjeu essentiel c'est comment avoir des bons politiques pour se protéger contre les attaques, pour cela nous préférons de les détecter au plus vite afin d'agir efficacement et de minimiser les dégâts.

C'est la raison pour laquelle un nouveau concept appeler système de détection d'intrusion a été introduit comme une seconde ligne de défense qui surveille et analyse les événements du système dans le but de trouver et de fournir un avertissement en temps réel ou quasi-réel des tentatives d'accès aux ressources du système d'une manière non autorisée.

1-Les systèmes de détection d'intrusion :

Avant de définir un système de détection d'intrusion, il est commode de définir la notion d'intrusion.

L'intrusion est souvent définie comme une pénétration illégale au système, une tentative d'un utilisateur du système d'obtenir des privilèges non autorisés, ou bien toute tentative de compromettre les services standards de la sécurité : la confidentialité, l'intégrité ou la disponibilité des informations.

1.1Définition :

1-Un système de détection d'intrusions (« Intrusion Détection System » ou IDS) est un appareil ou une application qui alerte l'administrateur en cas de faille de sécurité, de violation de règles ou d'autres problèmes susceptibles de compromettre son réseau informatique. [1]

2-Un IDS (Intrusion Detection System) un mécanisme écoutant le trafic réseau de manière furtive afin de repérer des activités anormales ou suspectes et permettant ainsi d'avoir une action de prévention sur les risques d'intrusion. [2]

3-Une application logicielle qui peut être implémentée sur des systèmes d'exploitation hôtes ou en tant que périphériques réseau pour surveiller l'activité associée à des intrusions ou à des abus d'initiés, ou les deux. [3]

4-Un système de sécurité pour les ordinateurs et les réseaux qui peuvent permettre l'inspection de l'activité des systèmes et de l'activité du réseau entrant/sortant. La fonction clé IDS identifie une activité ou des schémas suspects pouvant indiquer une attaque de réseau ou de système. Les IDS sont des outils permettant de détecter les attaques/intrusions du réseau sur lequel il est placé. C'est un outil complémentaire aux firewall, scanneurs de failles et anti-virus. [4]

5- IDS sont des systèmes capables de surveiller en temps réel un système et de détecter qu'une séquence d'actions est symptomatique d'un comportement anormal ou malicieux. De manière plus macroscopique, ces systèmes analysent les événements du système à protéger et détectent les attaques ou comportements illégitimes. Les IDS se satisfont d'un système d'alarmes, qui a pour but de prévenir le responsable qu'une attaque est en cours. Leur rôle est alors de proposer une vue synthétique de la sécurité du système et d'offrir des informations d'aide à la décision, pour que les responsables puissent agir en connaissance de cause. [5]

2-L'architecture des IDS :

2.1-Les Systèmes Hiérarchiques :

Une architecture hiérarchique distribuée représentée à La figure

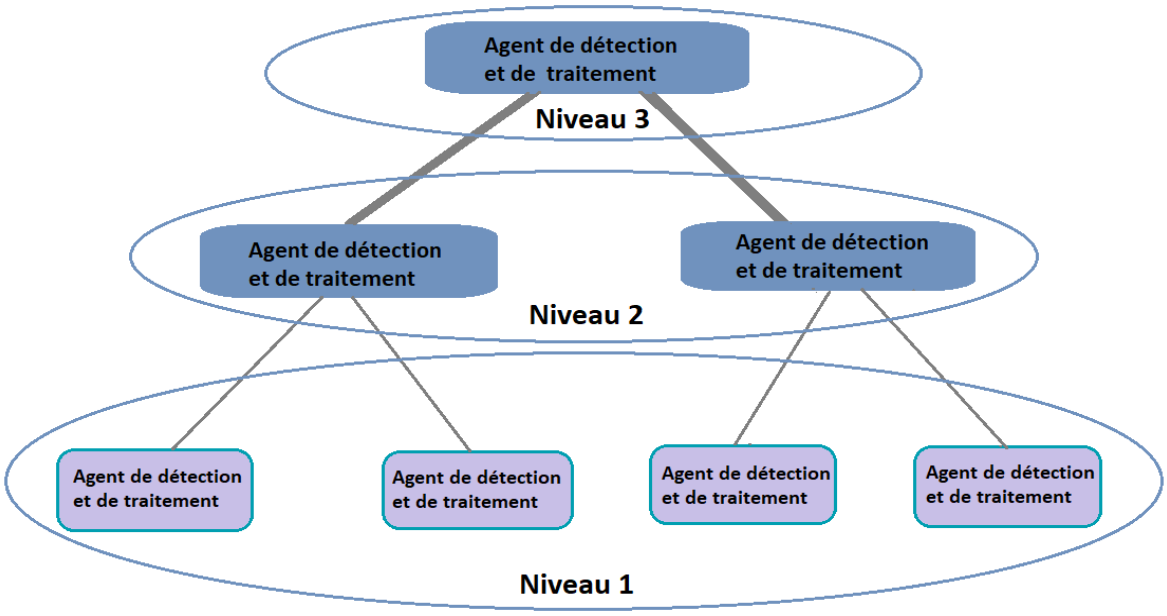


Figure 1: Architecture hiérarchique

Dans ce type, La détection des attaques et la collecte d'information a lieu aux extrémités de l'arbre. L'information est envoyée à un nœud interne qui rassemble l'information provenant de plusieurs nœuds. L'information est ainsi agrégée, abstraite et réduite à des niveaux supérieurs dans la hiérarchie pour éventuellement atteindre le nœud racine de l'arbre. Le nœud racine effectue l'analyse de l'information et prend des décisions quant à la nature de la menace et prépare une réponse appropriée, habituellement, l'unité de gestion et de contrôle avisent un opérateur humain de la situation pour que ce dernier puisse manuellement prendre action. [6]

Exemple de système hiérarchique : GrIDS, AAFID

Les avantages :

La principale force de ce système est de posséder un degré acceptable d'expansion avec un point central d'administration.

Les inconvénients :

- Si le nœud racine de l'arbre vient à être compromis ou mis hors service, alors tout le système est compromis.
- Utilisation de beaucoup de bande passante pour transmettre les messages de l'extrémité vers l'unité centrale.
- Le déploiement de ce type de Système à grande échelle est limité.

2.2-Système Peer-to-Peer :

Cette architecture sollicite les manques des systèmes hiérarchiques, elle repose sur les réseaux Peer-To-Peer. Le protocole sert que chaque hôte opère un IDS local et un gestionnaire de sécurité qui peut échanger des informations à propos des menace grâce à un système d'échange de message avec les autres hôtes. Dans ce genre de système chaque entité est responsable de sa propre sécurité et surveille ses voisins.

Avantage :

- Résolution des problèmes engendrés par l'approche hiérarchique.

Inconvénients :

- La difficulté de s'adapter aux changements qui peuvent se produire dans le réseau et aux comportements des utilisateurs qui varient considérablement.
- La difficulté de mise à jour de ces systèmes, lorsque l'on veut améliorer ou rajouter de nouvelles méthodes de détection. [4]

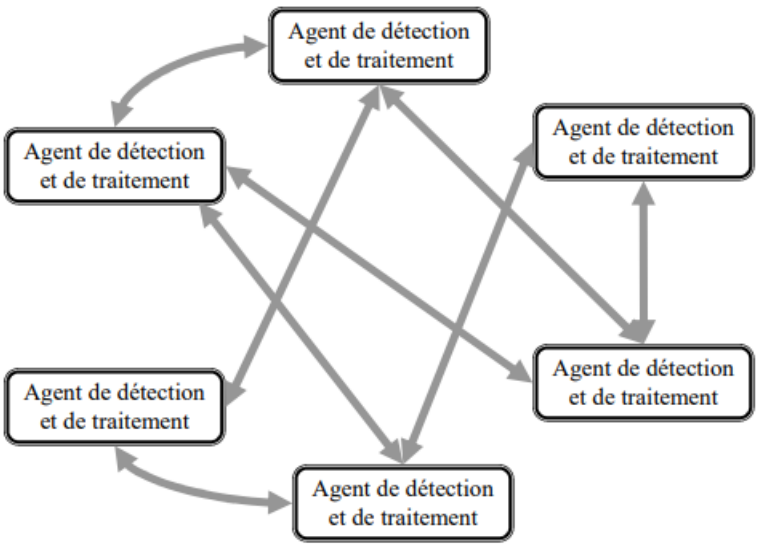


Figure 2:Architecture P2P

2.3-Système multi-agents :

Le système de détection d'intrusions doit pouvoir s'adapter à un environnement complexe, d'une part son évolution et sa variation continuelle, en termes de comportements utilisateurs et de problèmes de sécurité. La connaissance manipulée par le système de détection d'intrusions varie donc constamment et cette dynamité complexifie la gestion de la sécurité des réseaux. D'autres part, le système de gestion de sécurité doit respecter des contraintes temporelles et par conséquent réagir au plutôt lorsque des événements indiquent un état anormal du réseau (exemple une congestion du réseau due à une attaque de déni de service). [7]

Le SMA est composé de trois types de sondes : des agents, des récepteurs et des moniteurs. Les agents analysent les événements locaux et génèrent des rapports d'alerte lorsqu'une intrusion est détectée. Ces rapports sont envoyés aux récepteurs, qui supervisent les agents et corrèlent les informations. Les moniteurs reçoivent des rapports synthétiques de la part des récepteurs et peuvent corréler les rapports de plusieurs récepteurs. [8]

Les avantages :

- Les fonctionnalités et la sécurité des IDS sont décentralisées de façons à éviter les goulots d'étranglements au niveau du calcul et pour éviter la création d'un point d'échec unique.
- Ils permettent de détecter les intrusions au lieu où elles se produisent.
- Ils permettent une grande flexibilité et adaptation aux changements d'environnement.

Les inconvénients :

- Les communications entre agents doivent être surveillées et authentifiées de façons à s'assurer de l'intégrité et de la confidentialité des communications entre agents.
- Une plate-forme d'agents compatible doit être présente sur les hôtes pour recevoir les agents.
- La plate-forme et les agents mobiles peuvent être victime d'attaque.

3-Les différents types de IDS :

La première caractéristique de ces IDS est leur emplacement dans la structure à surveiller. Qu'ils s'agissent d'IDS placés en coupure sur le réseau ou directement sur chaque machine de l'infrastructure, le placement de ces systèmes implique un certain nombre de conséquences, telles que la capacité ou non de détecter certains types d'attaques, la responsabilité du déploiement ou de mise à jour de ces systèmes, etc. Nous détaillons dans les paragraphes suivants chacun des placements possibles.

3.1-Les systèmes de détection d'intrusion de type hôte (HIDS) :

Un HIDS se trouve sur la machine, et de ce fait il analyse l'activité se passant sur cette machine. Il analyse en temps réel les flux relatifs à une machine ainsi que ses journaux (logs). Un HIDS est entièrement opérationnel sur un système sain où il vérifie l'intégrité des données. Si le système a été compromis par un pirate, le HIDS ne sera plus efficace.

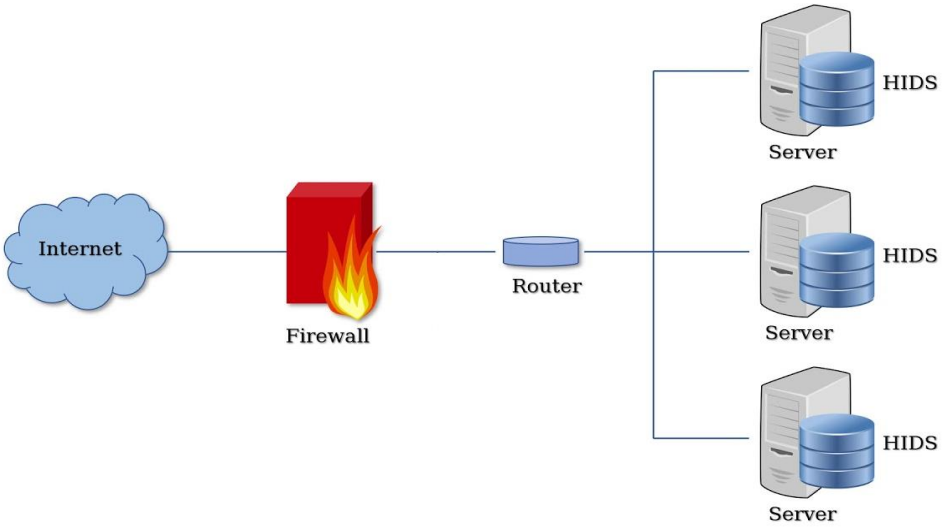


Figure 3 : Les HIDS

Exemple de HIDS :

- DarkSpy
- FCheck
- IceSword

3.2-Les systèmes de détection d'intrusions réseau (NIDS) :

Ils analysent les flux transitant sur le réseau de manière passive et détectent les intrusions en temps réel. Le trafic réseau est généralement (en tout cas sur Internet) constitué de datagrammes IP. Un N-IDS est capable de capturer les paquets lorsqu'ils circulent sur les liaisons physiques sur lesquelles il est connecté. Un N-IDS consiste en une pile TCP/IP qui réassemble les datagrammes IP et les connexions TCP.

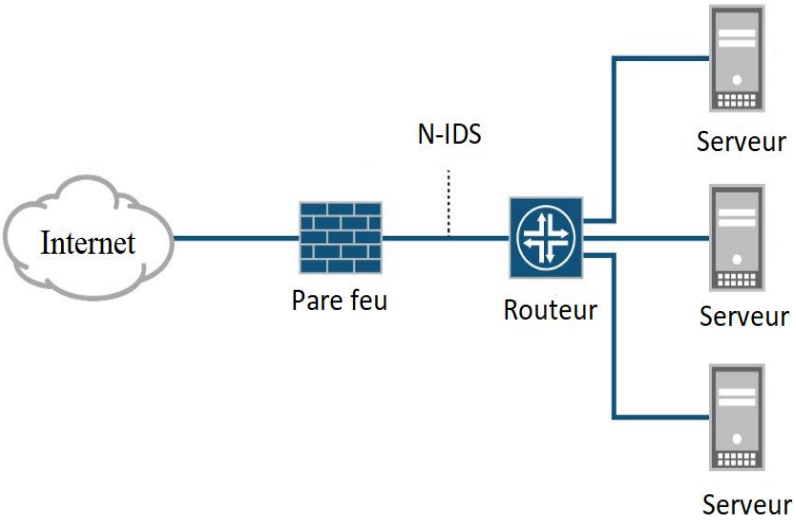


Figure 4 : Les NIDS

Exemple de NIDS :

- Snort
- Checkpoint
- Tipping Point

3.3- Les systèmes de détection d'intrusions hybrides :

Ces IDS permettent de réunir des informations de diverses sondes placées un peu partout sur le réseau. On dit qu'ils sont « hybrides » du fait qu'ils sont capables de réunir aussi bien des informations provenant de systèmes HIDS que NIDS.

Exemples de IDS hybrides :

- Prelude
- OSSIM
- Comparaison entre les deux types

4-Méthodes de détection :

Il existe deux méthodes de détection, la première consiste à utiliser des connaissances accumulées sur les attaques puis les exploiter afin de prouver l'existence d'autres attaques. La seconde consiste à créer un modèle basé sur le comportement habituel du système et surveiller toute déviation de ce comportement. La première méthode est appelée approche par scénario et la seconde est l'approche comportementale. [9]

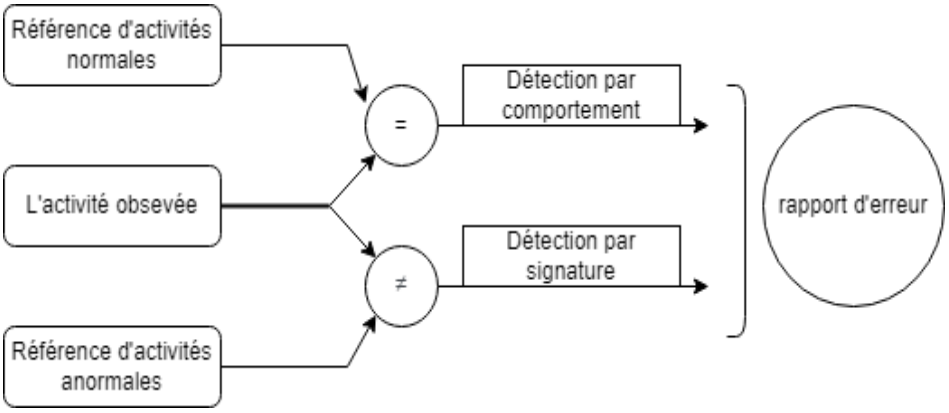


Figure 5: Méthode de détection d'IDS

4.1-L'approche par signature :

Les IDS basés sur les signatures (également appelés basés sur les définitions) utilisent une base de données de vulnérabilités connues ou de modèles d'attaque connus. Par exemple, des outils sont disponibles pour qu'un attaquant lance une attaque SYN flood sur un serveur en saisissant simplement l'adresse IP du système à attaquer. L'outil d'attaque inonde ensuite le système cible de paquets de synchronisation (SYN), mais ne termine jamais la poignée de main TCP (Transmission Control Protocol) à trois voies avec le paquet d'accusé de réception final (ACK). Si l'attaque n'est pas bloquée, elle peut consommer des ressources sur un système et finalement le faire planter.

Cependant, il s'agit d'une attaque connue avec un modèle spécifique de paquets SYN successifs d'une IP à une autre IP. L'IDS peut détecter ces modèles lorsque la base de données de signatures inclut les définitions d'attaque. Le processus est très similaire à celui utilisé par un logiciel antivirus pour détecter les logiciels malveillants. Vous devez mettre à jour régulièrement les signatures IDS et les définitions antivirus du fournisseur pour vous protéger contre les menaces actuelles.

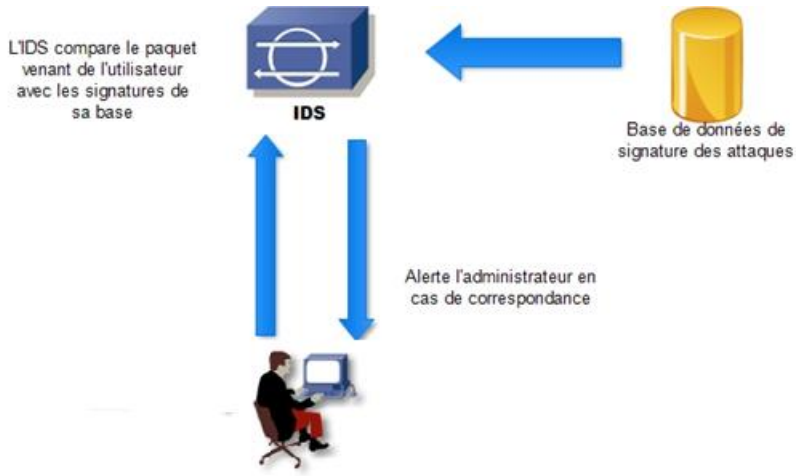


Figure 6:Approche par signature

Avantages :

- Les critères de signatures pouvant être précisément définis, il y a peu de fausses alertes.
- Classification facile de la criticité des attaques signalées.
- Facilité de mise à jour et évidemment dans la quantité importante de signatures contenues dans la base du NIDS.

Inconvénients :

- Si les signatures ne sont pas clairement définies, les attaques passent inaperçues.
- Nécessite de mettre à jour régulièrement la table des attaques connues.

4.2-L'approche comportementale :

La détection basée sur les anomalies (également appelée basée sur l'heuristique ou basée sur le comportement) identifie d'abord le fonctionnement normal ou le comportement normal. Pour ce faire, il crée une base de performances dans des conditions de fonctionnement normales. L'IDS fournit une surveillance continue en comparant constamment le comportement actuel du réseau par rapport à la ligne de base. Lorsque l'IDS détecte une activité anormale (en dehors des limites normales identifiées dans la ligne de base), il émet une alerte indiquant une attaque potentielle.

La détection basée sur les anomalies est similaire au fonctionnement d'un logiciel antivirus basé sur l'heuristique. Bien que les méthodes internes soient différentes, elles examinent l'activité et prennent des décisions qui sortent du cadre d'une base de données de signatures ou de définitions. Cela peut être efficace pour découvrir des exploits zero-day. Une vulnérabilité zero-day est généralement définie comme une vulnérabilité inconnue du fournisseur. Cependant, dans certaines utilisations, les administrateurs définissent un exploit zero-day comme un exploit pour lequel le fournisseur n'a pas publié de correctif. En d'autres termes, le fournisseur peut être au courant de la vulnérabilité mais n'a pas encore écrit, testé et publié de correctif pour fermer la vulnérabilité.

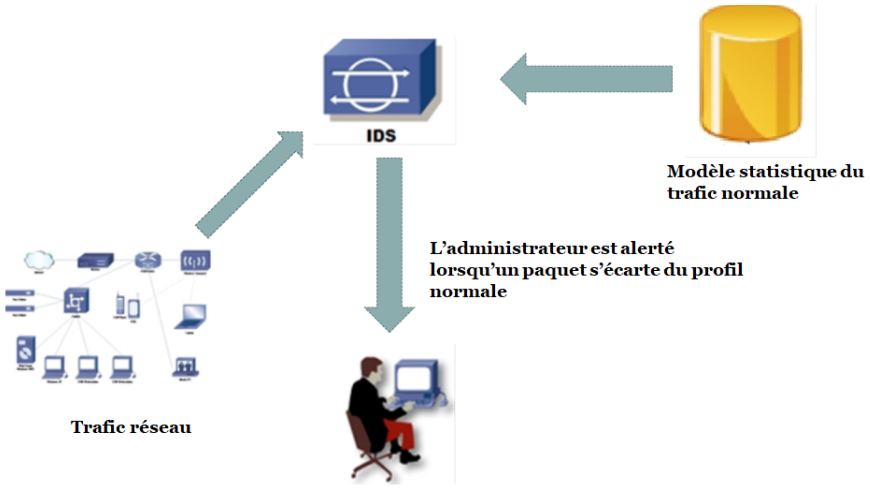


Figure 7:Approche comportementale

Avantage :

- Découverte de nouvelles techniques d'attaque.

Inconvénients :

- Difficulté à créer un profil précis de l'utilisateur.
- Risque de nombreuses fausses alarmes.

5-Comportement après détection :

Si l'IDS détecte une attaque, deux comportements peuvent être adoptés : une réponse passive ou bien une réponse active. Cet aspect est souvent lié au module de réponses de l'IDS.

5.1- Réponse passive :

Dans ce cas, la réaction de l'IDS se limite à une alerte d'identification de l'attaque envoyée à l'administrateur ou bien vers un système d'archivage (fichiers logs). Dans les deux cas c'est l'opérateur humain qui va se charger des contre-mesures.

5.2- Réponse active :

Inversement au premier cas, des contre-mesures automatiques seront actionnées pour contrer l'attaque et limiter sa portée. Par exemple, bloquer en entrée des adresses IP ou des ports, fermer une session ou bien arrêter une machine.

Conclusion

Dans ce chapitre nous avons détaillé les systèmes de détection des intrusions vu que c'est notre objectif dans ce mémoire, qui jouent un rôle complémentaire aux mécanismes de sécurité traditionnels. Nous avons présenté aussi le principe de fonctionnement des IDS ainsi leurs architectures et leurs classifications selon différents critères avec leurs avantages et inconvénients, parmi les critères de classification abordés la méthode de détection qui divise les IDS en deux types, les IDS comportementaux et à base de signatures, et pour enrichir notre recherche nous allons s'appuyant sur les graphes de connaissance, qui sont l'objet des prochains chapitres.

Chapitre2

Graphes structurés des données

Introduction

Les données graphiques sont devenues omniprésentes au cours de la dernière décennie il est devenu plus difficile de trouver ou d'obtenir des informations et des connaissances précieuses à partir de ces énormes données bruyantes, grâce à la recherche et à l'expérience, un ensemble de modèles et de pratiques appelés graphes de connaissances (knowledge graph) a été développé pour soutenir l'extraction de connaissances à partir de données.

Les graphes de connaissances sont utilisés pour cartographier la collecte de données provenant de différentes sources et créer une connexion entre les différentes entités d'un sujet donné pour donner un sens aux données et supprimer toute ambiguïté sémantique.

1-Concepts et définitions :

1.1-Un graphe : est un schéma contenant des points nommés sommets, reliés ou non par des segments appelés arêtes ou arcs.

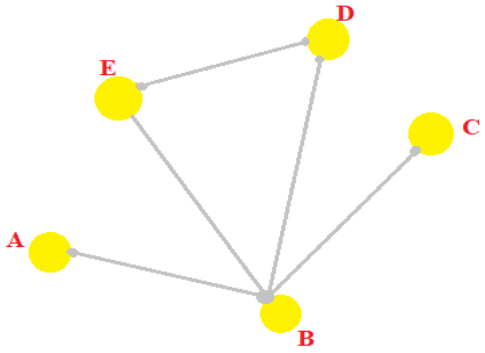


Figure 8: Représentation d'un graphe

A est un sommet, le segment [AB] est une arête reliant A à B (ou B à A). [10]

1.2-Une base de données : est un lot d'informations stockées dans un dispositif informatique. Les technologies existantes permettent d'organiser et de structurer la base de données de manière à pouvoir facilement manipuler le contenu et stocker efficacement de très grandes quantités d'informations.

1.3-Un réseau graphique : se compose de nœuds représentant des objets et d'arcs qui décrivent la relation entre ces objets.

1.4-La représentation des connaissances : est définit comme une notion mieux décrite par les cinq rôles distincts qu'elle joue, parmi lesquelles, il s'agit d'une « théorie fragmentaire du raisonnement intelligent » qui s'exprime comme « l'ensemble des inférences que la représentation (l'entité) sanctionne ». En d'autres termes, la connaissance d'une entité est représentée par l'entité elle-même et les relations déduites qu'elle entretient avec d'autres entités, faits, circonstances … etc.

Il existe quatre techniques principales de représentation des connaissances : logique, sémantique, cadre et règles de production. [11]

1.5-L'ontologie : une ontologie décrit formellement les types, les propriétés et les interrelations entre les entités, elle peut être définit comme un ensemble d'axiomes qui peuvent être considérés comme des principes qui définissent les connaissances dans un domaine particulier. [11]

2-Origines des graphes de connaissances (Historique) :

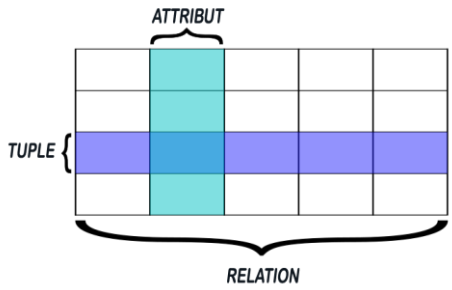
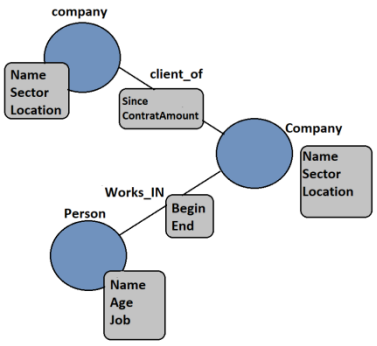
Le terme de « graphe de connaissance » ou « Knowledge Graph » existe depuis des décennies introduit par Google, puis par un nombre grandissant d'autres entreprises, l'ont rendu extrêmement populaire dernièrement. De plus son couplage avec différentes techniques d'intelligence artificielle contribue à en faire un sujet d'intérêt d'actualité. A l'instar de cette expression « intelligence artificielle », le terme « graphe de connaissance » est utilisé avec différentes considérations et identifie actuellement une ressource numérique très différente d'un cas d'usage à un autre, le domaine de la représentation des connaissances à base de graphes existe depuis longtemps et étudie l'expressivité de ces modèles et la complexité de leurs traitements avec des interactions multidisciplinaires et des applications dans de nombreux domaines. [12]

3- Les graphes de connaissances :

Une base de données graphe est fondamentalement différente de base de données relationnelle traditionnelle.

Une base de données de graphe rend possibles différentes utilisations de la donnée, et elle peut mener à penser différemment à propos de la donnée.

Voici un tableau comparatif entre la base de données relationnelle et la base de données graphe. [13]

Base de données relationnelle	Base de données graphe
	Model graph 

Les relations entre les tables sont implicites. Elles sont définies à même la structure de la base de données par le biais d'un index et de clés.	Les relations entre les objets sont explicites. Chaque arc désigne une relation précise et cette relation constitue une entité à part entière dans la base de données au même titre que les nœuds.
Linéaire ou hiérarchique. Elle gère très bien les relations un-à-un ou un-à-plusieurs entre les occurrences de la base de données. Elle a cependant plus de difficulté avec les relations plusieurs-à-plusieurs.	Multidirectionnelle. Elle excelle avec les relations plusieurs-à-plusieurs, chaque nœud pouvant avoir un grand nombre de relations avec plusieurs autres nœuds. Un nœud peut être plusieurs choses à la fois (tout comme un être humain).
Efficace pour protéger des données. Une base de données relationnelle peut néanmoins échanger des données avec une autre base de données par le biais d'une API.	Efficace pour exposer et échanger des données. La base de données graphe permet de désigner des objets et des relations selon la Resource Description Framework (RDF), les bases de données graphe suivant la spécification RDF peuvent être exposées sous forme de données ouvertes liées, lesquelles peuvent être aisément liées à d'autres bases de données graphe RDF.
Permet d'accumuler beaucoup de données du même type.	Flexible. La base de données graphe peut facilement être adaptée pour recevoir de nouveaux types de données dès que le besoin s'en fait sentir.

Tableau 1:comparaison entre une BD relationnelle et BD graphique

3.1- Définition du graphe de connaissances :

1- Un graphe de connaissances est un graphe dirigé étiqueté dans lequel les étiquettes ont des significations bien définies, se compose de nœuds, d'arêtes et d'étiquettes. Un bord relie une paire de nœuds et capture la relation d'intérêt entre eux, par exemple une connexion réseau entre deux ordinateurs. Les étiquettes capturent le sens de la relation.

Plus formellement, étant donné un ensemble de nœuds **N** et un ensemble d'étiquettes **L**, un graphe de connaissances est un sous-ensemble du produit croisé **N × L × N**. Chaque membre de cet ensemble est appelé un triplet et peut être visualisé comme indiqué ci-dessous. [14]

Figure 9: Les composants d'un graphe de connaissance

À la base, le jeu final de toute exploration de connaissances est une liste d'entités (c'est-à-dire une séquence reconnaissable de caractères avec une signification spécifique) et de triplets ; souvent simplement décrit comme un **sujet, un objet et un prédicat** (ou entité-attribut-valeur et bien d'autres variantes…).

Prenons un autre exemple plus détaillé d'un graphe de connaissance comme le montre la figure 10

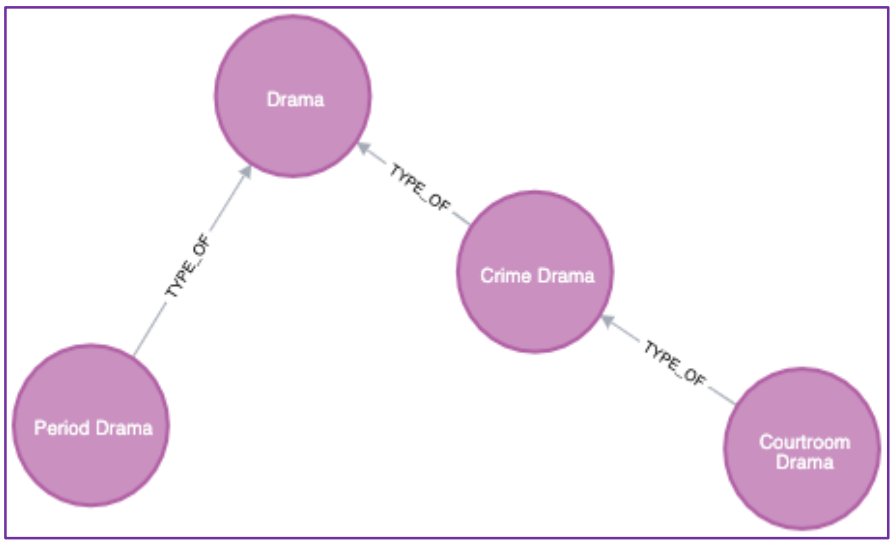


Figure 10: Un simple KG du genre dramatique pour les films

2- Un graphe de connaissances, également connu sous le nom de réseau sémantique, représente un réseau d'entités du monde réel, c'est-à-dire des objets, des événements, des situations ou des concepts, et illustre la relation entre eux. Ces informations sont généralement stockées dans une base de données de graphes et visualisées sous la forme d'une structure de graphe. [15]

3- Un graphe de connaissances est composé de trois composants principaux : les nœuds, les arêtes et les étiquettes. Tout objet, lieu ou personne peut être un nœud. Une arête définit la relation entre les nœuds. [16]

3.2-Exemples de graphes de connaissances :

Différentes entreprises ont développé différents types de graphes de connaissances qui sont déployés à des fins diverses, par exemple, certaines entreprises utilisent des graphes de connaissances internes ou plus petits pour la fonction en ligne et bien d'autres.
Le tableau ci-dessous répertorie certains des plus grands graphes de connaissances ainsi que leurs fonctions et leurs développeurs.

	Développeur	**Objectif et fonction**
1	Microsoft	Déployez le graphe de connaissances pour le moteur de recherche Bing, les données LinkedIn et les universitaires.
2	EBay	À partir de maintenant, construire un graphe de connaissances qui donne les relations entre les utilisateurs et les produits, fourni sur le site Web.
3	Google	Le graphe de connaissances de Google s'est largement adopté dans le cadre d'une fonction de catégorisation volumineuse sur les appareils de Google et rapidement intégrée dans le moteur de recherche
4	Facebook	Pour établir des liens entre les personnes, les événements et les idées, et se concentrer essentiellement sur les nouvelles, les personnes et les événements associés au réseau social.
5	IBM	Rend un cadre pour diverses entreprises/industries afin de générer des graphes de connaissances internes.

Tableau 2:Exemple de graphe de connaissance

4-La construction des graphes de connaissance :

Ce fait en deux parties :

4.1-Collecte et extraction de l'information :

Les types de données sont multiples et incluent les textes, les données intégrées mais aussi la vidéo et l'audio. Pour l'extraction d'informations nécessaires à la construction du graphe à partir de textes non structurés, sur un besoin de techniques de traitement automatique du langage ou NLP (Natural Language Processing) :

- Extraction des entités/nœuds
- Extraction des relations.
- Une fois les entités et relations extraites, on a intégré le graphe en intégrant les concepts et les contraintes définis par l'ontologie.

4.2-Vérification et déduction :

Une étape de vérification est nécessaire afin de détecter les incohérences dans le graphe. La dernière étape de la construction d'un graphe de connaissance consiste à inférer de nouvelles relations entre les nœuds sur la base des relations déjà présentes dans le graphe. Une première approche est d'utiliser l'inférence logique. Néanmoins cela peut s'avérer très vite ingérable car il faut pouvoir définir une ou plusieurs règles pour chaque type de relation. Une autre approche consiste à utiliser l'apprentissage automatique ou machine Learning.

La création de graphes de connaissance est complexe et présente quelques challenges :

- La qualité des données, pour garantir la qualité d'un graphe de connaissance il faut :
    - Assurer la mise à jour permanente des données ;
    - S’assurer que les données sont correctes ;
    - Assurer que les données sont complètes ;
- La vérification et l'enrichissement
    - Il faut pouvoir détecter les doublons
    - Il faut pouvoir gérer les conflits ;
- L'établissement de contraintes sur les relations [17]

5- L’architecture d’un graphe de connaissance :

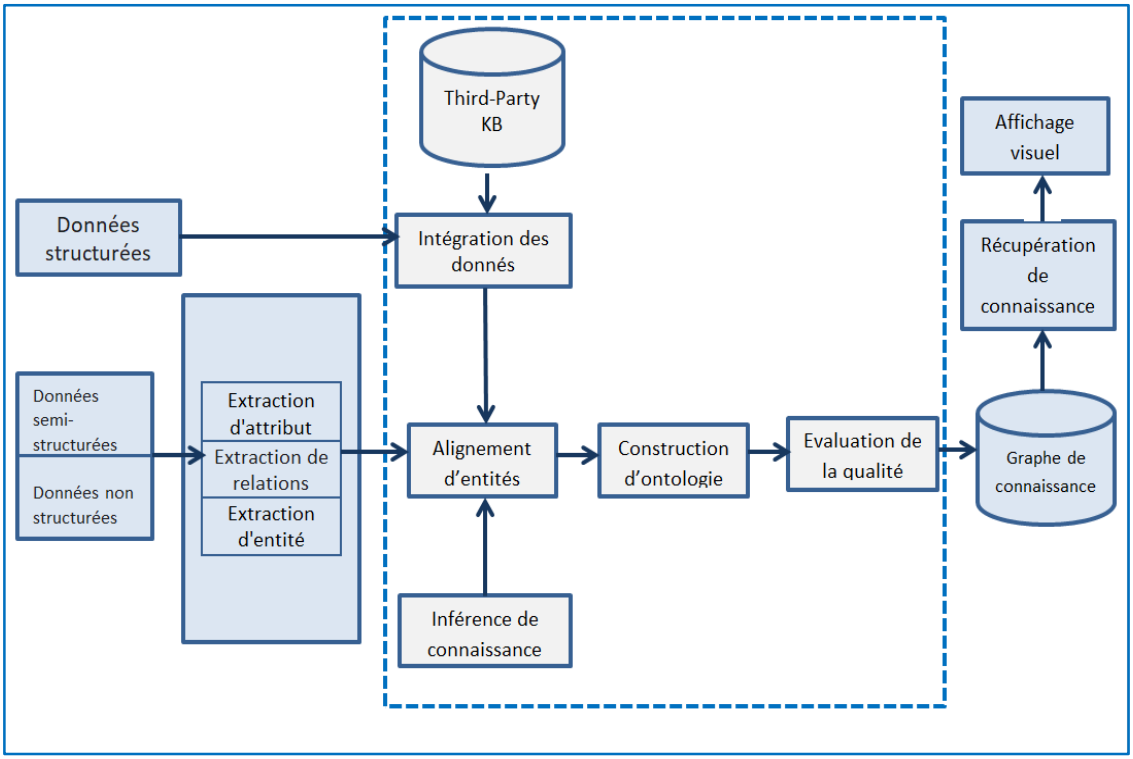


Figure 11: Architecture d'un graphe de connaissance

-Sources de connaissances : y compris les données structurées, les données non structurées et les données semi-structurées.

-Extraction d’informations : il s'agit d'extraire des entités, des attributs et des relations entre des entités de divers types de sources de données, et de former la représentation des connaissances de l'ontologie sur cette base. Il y a une grande quantité de données non structurées ou semi-structurées dans le processus de construction du graphe de connaissances. Dans le processus de construction du graphe de connaissances, ces données doivent être extraites par des méthodes de traitement du langage naturel. À partir de ces données, nous pouvons extraire des entités, des relations et des attributs.

-Fusion de connaissances : le travail principal consiste à extraire des données structurées et des informations d'entité extraites d'informations, et même des bases de connaissances tierces pour effectuer l'alignement des entités et la désambiguïsation des entités. Le résultat de cette étape doit être diverses informations d'ontologie fusionnées à partir de diverses sources de données.

-Traitement des connaissances : Le travail important dans le raisonnement des connaissances est l'achèvement du graphe des connaissances. Les méthodes couramment utilisées pour compléter les graphes de connaissances comprennent : des méthodes complémentaires basées sur le raisonnement ontologique, la mise en œuvre de mécanismes de raisonnement associés

et des méthodes complémentaires basées sur la structure du graphe et les caractéristiques du chemin de relation. [18]

6- Modèles de graphes de connaissances :

Il existe deux modèles courants de bases de données graphes : les graphes du Resource Description Framework (RDF) et les graphes de propriétés.

6.1- RDF (Resource Description Framework) :

Il s'agit plutôt d'un modèle de données pour décrire des ressources sur le web en spécifiant des triplets qui décrivent la logique de la modélisation d'un graphe de connaissance.

Les triplets prennent la forme comme suit : **<Sujet> <Prédicat> <Objet>.**

Pour des raisons historiques, l'univers RDF est communément classé en deux types de triplets. Ceux qui se rangent dans la boîte **T** (pour Terminologique qui sont les ontologies, les classes et les règles) et ceux qui se rangent dans la boîte **A** (pour Assertionnelle qui sont les instances de classes).

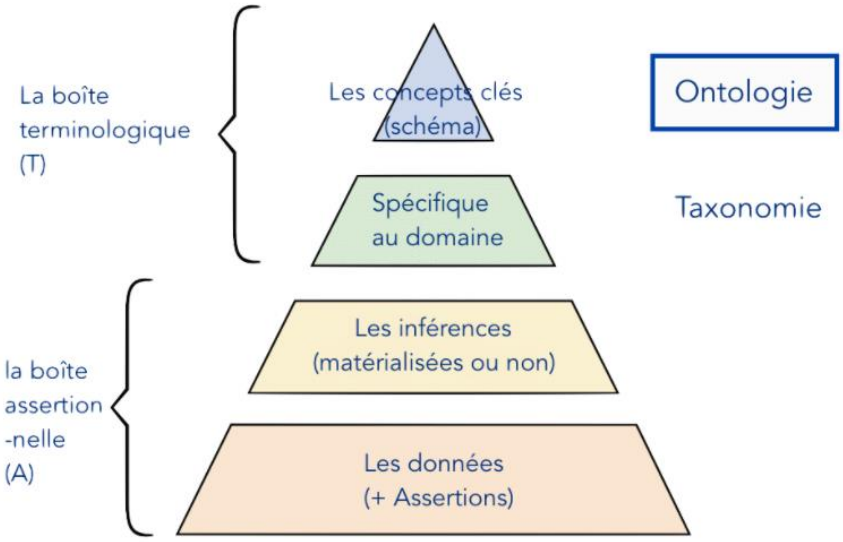


Figure 12: Architecture d'un graphe de connaissance RDF

6.2- Modèle de données de graphe de propriétés :

Le modèle de graphe de propriétés est le modèle le plus populaire dans les bases de données de graphes modernes, il implique une méthode populaire pour créer des graphes de connaissances. Il se compose des éléments suivants :

1-Les Nœuds qui représentent les entités dans le domaine

• Les nœuds peuvent contenir aucune propriété ou plusieurs, qui sont des paires clé-valeur qui représentent des données d'entité.

• Les nœuds peuvent avoir aucune étiquette ou plusieurs, indiquant l'objectif dans le graphique.

2-Les Relations qui représentent la façon dont les entités interagissent. Elle se caractérisent par :

• Le type de la relation.

• Chaque relation a un sens, allant d'un nœud à un autre.

• Les relations peuvent contenir aucune propriété ou plusieurs, qui sont des paires clé-valeur qui représentent certaines caractéristiques du lien telles qu'un horodatage ou une distance.

• Il y a toujours un nœud de début et un nœud de fin (qui peuvent être le même nœud).

Ces primitives (nœuds, relations et propriétés) et ces règles peuvent être utilisées pour assembler des modèles de données graphiques sophistiqués et haute-fidélité avec une relative facilitée. [19]

7- Application des graphes de connaissance :

Les applications des graphes de connaissance sont multiples, ils sont notamment importants pour des applications d'intelligence artificielle qui nécessitent de comprendre le langage humain en leur apportant une dimension sémantique :

7-1. Système de réponse aux questions « question answering » :

Les graphes de connaissance sont utilisés pour répondre aux questions exprimées en langage naturel

7-2. Systèmes de recommandation :

Le filtrage collaboratif est un type de système de recommandation qui effectue des recommandations en fonction des préférences communes des utilisateurs et des interactions historiques.

En général, l'utilisation des KG dans les systèmes de recommandation permet d'améliorer la précision et d'augmenter la diversité des éléments recommandés, ainsi que d'apporter une interopérabilité aux recommandations.

7-3. Récupération de l'information :

Aujourd'hui, de plus en plus de moteurs de recherche commerciaux basés sur le Web intègrent les données d'entité des KG pour améliorer leurs résultats de recherche. Par exemple, Google intègre les données de Google Plus et Google Knowledge Graph, tandis que Facebook utilise Graph Search.

La connaissance humaine des entités du monde réel dans les KG aide les moteurs de recherche en améliorant leur capacité à comprendre les requêtes et les documents. Une telle recherche orientée entité s'améliore avec le développement de KG à grande échelle. Les KG peuvent être utilisés dans différents composants tels que la représentation des requêtes, la représentation des documents et le classement d'un système de recherche. [20]

8-Les algorithmes de Graphe de connaissance :

Les graphes de connaissance ont de nombreux algorithmes. Les plus connus se répartissent en 5 catégories :

- Détection communautaire.
- Centralité.
- Prédiction.
- Trouver son chemin.
- Similarité.

Conclusion

Ce chapitre a été consacré pour les graphes de connaissances, nous avons présenté en premier lieu une introduction au domaine des graphes de connaissance, où nous avons abordé les notions de base et les définitions essentielles relatives. Nous avons notamment insisté sur son architecture et ses domaines d'application et finalement nous avons parlé des algorithmes de graphe de connaissance, ces derniers font encore l'objet de notre recherche.

Chapitre 03

Machine Learning

Introduction

L'objectif de la recherche en intelligence artificielle (IA) est de doter un système informatique de capacités de réflexion similaires à celles des humains. Il y a donc un défi dans la compréhension du raisonnement humain.
Cela a permis de réaliser d'importants progrès dans la dernière décennie, et dans différents secteurs. Les avancées les plus connues sont celles réalisées dans Le machine Learning ou **« l'apprentissage automatique »** qui est une branche de l'Intelligence artificielle.

Dans ce chapitre les types de système d'apprentissage automatique seront approchés. S'ensuivra des différents algorithmes utilisés dans le cadre du machine Learning.

1-Définitions :

1.1-Intelligence artificielle :

L'IA désigne la possibilité pour une machine de reproduire des comportements liés aux humains, tels que le raisonnement, la planification et la créativité.
L'IA permet à des systèmes techniques de percevoir leur environnement, gérer ces perceptions, résoudre des problèmes et entreprendre des actions pour atteindre un but précis.

Les systèmes dotés d'IA sont capables d'adapter leurs comportements plus ou moins en analysant les effets produits par leurs actions précédentes, travaillant de manière autonome. [21]

2- L'apprentissage automatique (Machine Learning) :

1-Le machine Learning ou « apprentissage automatique » en français est un concept qui fait de plus en plus parler de lui dans le monde de l'informatique, et qui se rapporte au domaine de l'intelligence artificielle. Encore appelé « apprentissage statistique », ce terme renvoie à un processus de développement, d'analyse et d'implémentation conduisant à la mise en place de procédés systématiques. Pour faire simple, il s'agit d'une sorte de programme permettant à un ordinateur ou à une machine un apprentissage automatisé, de façon à pouvoir réaliser un certain nombre d'opérations très complexes. [22]

L'objectif visé est de rendre la machine ou l'ordinateur capable d'apporter des solutions à des problèmes compliqués, par le traitement d'une quantité d'informations.

L'Apprentissage automatique se décompose en deux étapes : une étape d'entraînement c'est l'apprentissage sur un ensemble de données et une deuxième étape de vérification c'est le teste.
Nous aurons donc trois phases : la représentation, l'évaluation et l'optimisation.
La phase de représentation consiste à trouver le modèle mathématique le plus adapté. L'évaluation elle mesure l'écart entre le modèle et la réalité des données de tests. Enfin, l'optimisation vise à amenuiser cet écart. [23]

3- Les types d'apprentissage automatique :

Nous pouvons classer les types d'apprentissage en plusieurs catégories très distinctes : apprentissage supervisé, non supervisé et apprentissage par renforcement, comme le montre la figure ci-dessous :

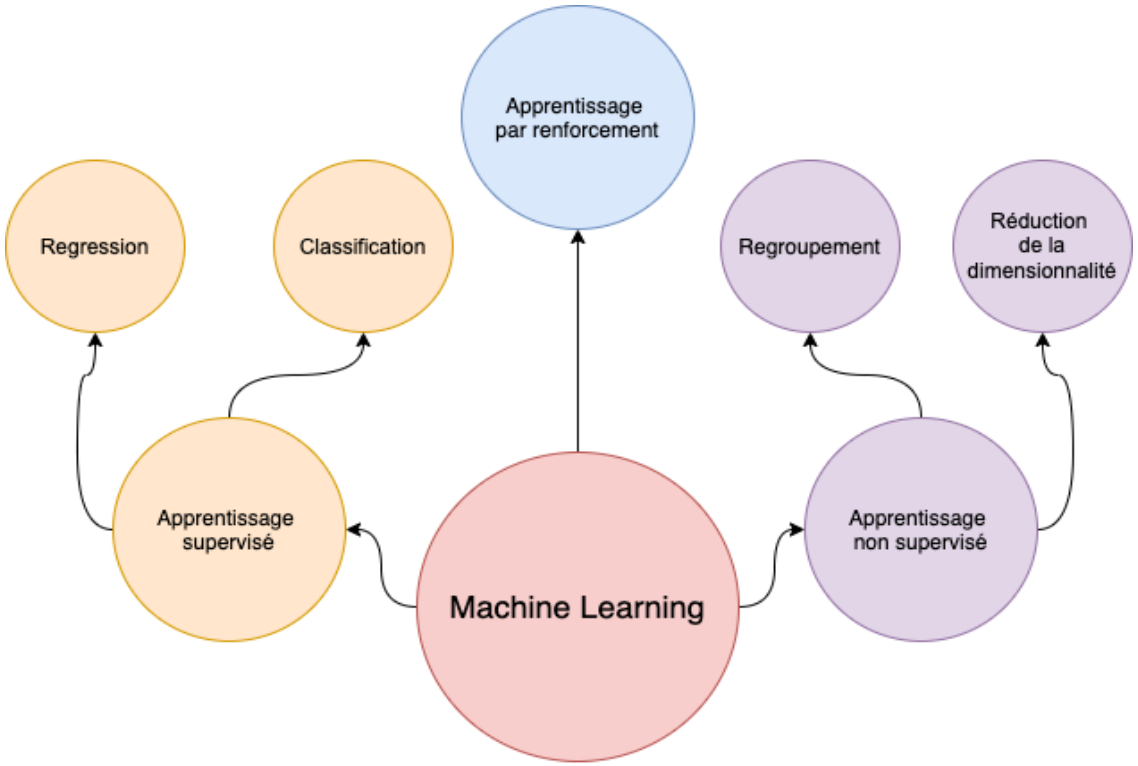


Figure 13: Les types d'apprentissage automatique

3.1-Apprentissage supervisé :

L'apprentissage supervisé consiste à entraîner un modèle en lui fournissant la réponse (label). Cette réponse permet de superviser l'apprentissage du modèle en lui disant à quel point il est loin de la bonne réponse. Dans un apprentissage supervisé, nous avons un X (variable indépendante) et un Y (variable dépendante) lors de l'entraînement. Cette catégorie se divise en deux sous catégories principales soit la classification et la régression.

1-La classification : survient lorsque la réponse que nous fournissons au modèle est une classe, et non une valeur continue.

2-La régression : survient lorsque la réponse que nous fournissons au modèle est une valeur numérique continue. [24]

En d’autres mots, ce qui différencie la classification de la régression est le type de sorties et non le type d’entrées. Si la sortie est une classe il s’agit d’une classification. Si la sortie est un intervalle, il s’agit d’une régression.

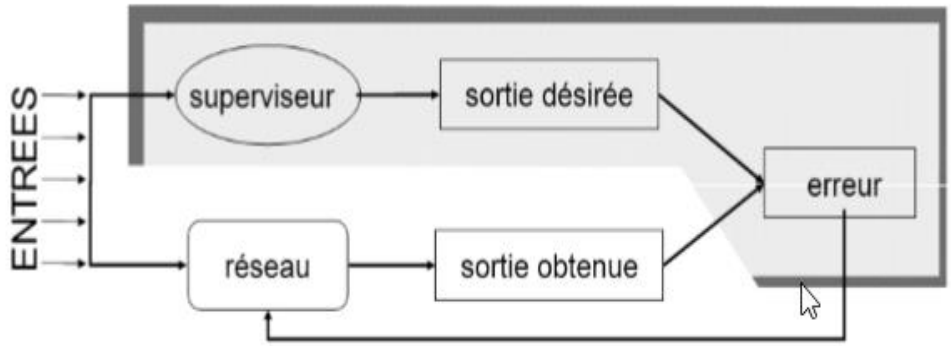


Figure 14: Schéma d’un modèle supervisé

3.2-Apprentissage non supervisé :

L’apprentissage non supervisé consiste à entraîner un modèle à trouver les caractéristiques et extraire les relations entre les données. Dans un problème non supervisé, nous n’avons pas la réponse exacte que le modèle devrait trouver, nous avons seulement des données entrantes.

En apprentissage non supervisé, nous avons seulement un X (variable indépendante) lors de l'entraînement. L’apprentissage non supervisé se divise lui aussi en deux principales catégories soit le regroupement (Clustering), soit la réduction de la dimensionnalité (dimensionality reduction).

1-Le regroupement : consiste à identifier les différents groupes dans un jeu de données.

2-La réduction de la dimensionnalité : est utilisée principalement de deux façons. Premièrement, elle peut être utilisée afin compresser des données, ce qui a pour effet d'utiliser moins de mémoire et d’espace disque ainsi que réduire le temps d'entraînement. Deuxièmement, elle peut être utilisée afin de visualiser des données. [21]

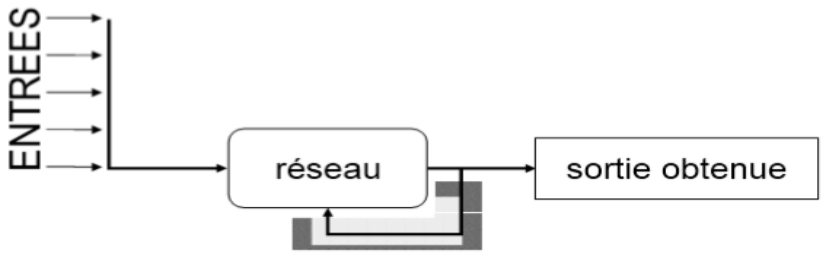


Figure 15: Schéma d’un modèle non supervisé

3.3- Apprentissage par renforcement :

L'apprentissage par renforcement consiste en un agent qui interagit avec son environnement.

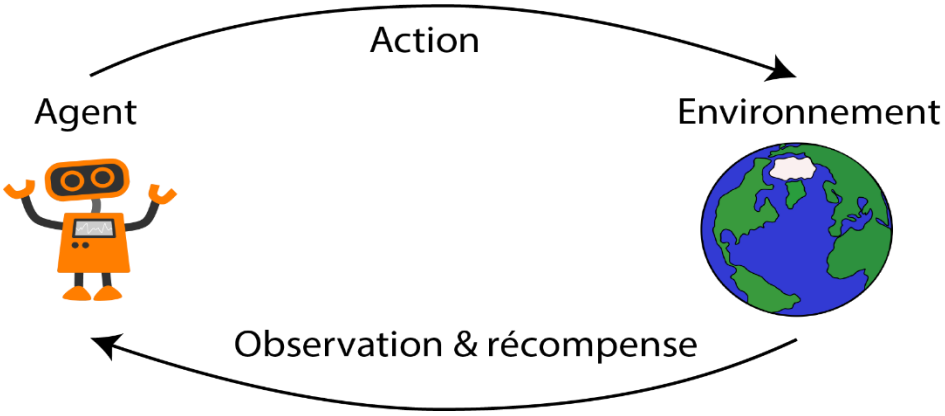


Figure 16: Apprentissage par renforcement

L'agent reçoit une observation et une récompense provenant de l'environnement. L'observation indique l'état de l'environnement à un moment précis et la récompense est un indicateur de la performance de l'agent. Par la suite, l'agent analyse l'observation et la récompense qu'il a reçue afin de décider quelle action il devrait prendre. [21]

4-Les algorithmes de machine Learning :

Le machine Learning offre un certain nombre d'algorithmes pour traiter des tâches de régression et de classification avec plusieurs variables dépendantes et indépendantes. Parmi ces algorithmes :

4.1- La régression linéaire (Linear Regression) :

Les algorithmes de régression linéaire modélisent la relation entre des variables prédictives et une variable cible**.** La relation est modélisée par une fonction mathématique de prédiction. [25]

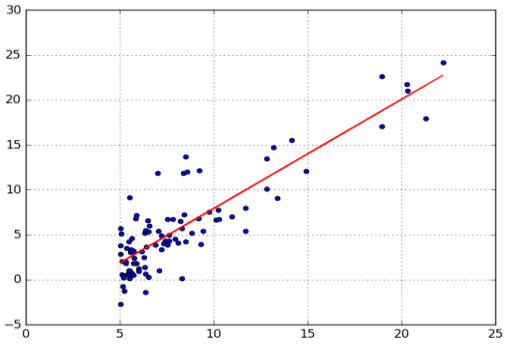


Figure 17: Schéma d'algorithme de régression linéaire

4.2- La régression logistique (Logistic Regression) :

La régression logistique est une méthode statistique pour effectuer des classifications binaires. Elle prend en entrée des variables prédictives qualitatives et/ou ordinales et mesure la probabilité de la valeur de sortie en utilisant la fonction sigmoïde (représentée dans la figure suivante).[22]

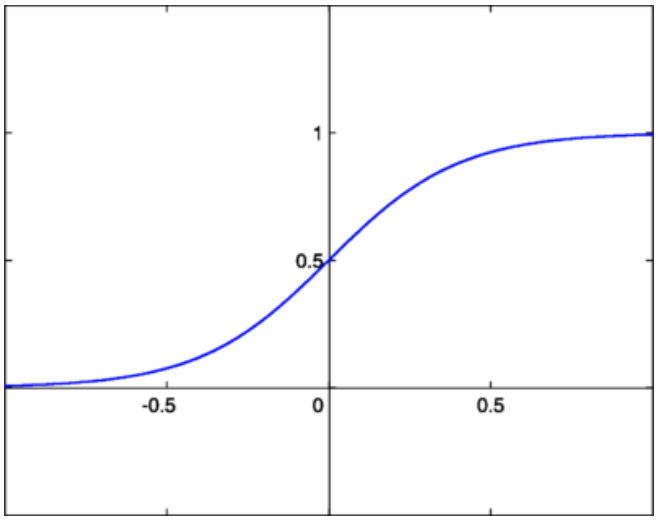


Figure 18: Schéma d'algorithme de régression logistique

Avantage :

-Le modèle est facile à interpréter

Inconvénients :

-Sensible aux bruits

-Négligence des interactions entre les variables prédictives

4.3- Les machines à vecteurs de support (SVM) :

SVM est l'un des algorithmes d'apprentissage supervisé les plus populaires, qui est utilisé pour les problèmes de classification et de régression.

L'objectif de l'algorithme SVM est de créer la meilleure ligne ou limite de décision capable de séparer l'espace à **n** dimensions en classes afin que nous puissions facilement placer le nouveau point de données dans la bonne catégorie à l'avenir. Cette frontière de meilleure décision est appelée un hyperplan.

SVM choisit les points/vecteurs extrêmes qui aident à créer l'hyperplan. Ces cas extrêmes sont appelés vecteurs de support. [26]

Considérez le diagramme ci-dessous dans lequel deux catégories différentes sont classées à l'aide d'une limite de décision ou d'un hyperplan :

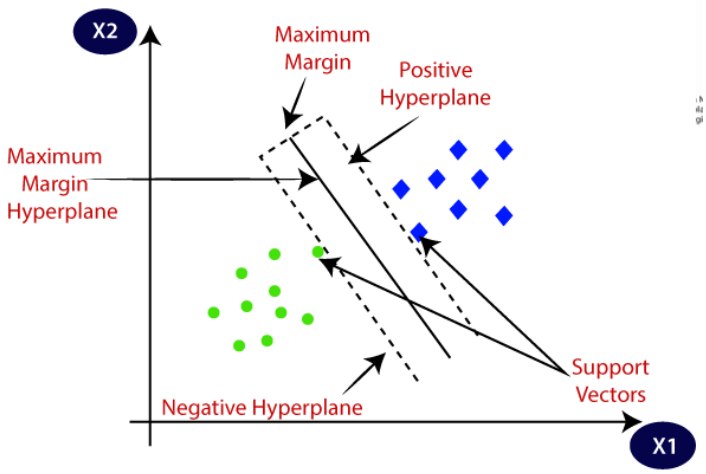


Figure 19: l'algorithme SVM

Avantages :

-Il permet de traiter des problèmes de classification non linéaire complexe.

-Les SVM constituent une alternative aux réseaux de neurones car plus faciles à entraîner.

Inconvénient :

- Les SVM sont souvent moins performants.

4.4- Naïve Bayes :

Naïve Bayes est un classifieur assez intuitif à comprendre. Il se base sur le théorème de Bayes des probabilités conditionnelles

$$P(A|B)P(B) = P(A \cap B) = P(B|A)P(A)$$

Naïve Bayes assume une hypothèse forte (naïve). En effet, il suppose que les variables sont indépendantes entre elles.
Généralement, le Naïve Bayes est utilisé pour les classifications de texte. [22]

4.5- L'arbre de décision (Decision Trees) :

L'arbre de décision est un algorithme qui se base sur un modèle de graphe (les arbres) pour définir la décision finale. Chaque nœud comporte une condition, et les branchements sont en fonction de cette condition (Vrai ou Faux). Plus on descend dans l'arbre, plus on cumule les conditions. La figure ci-dessous illustre ce fonctionnement. [22]

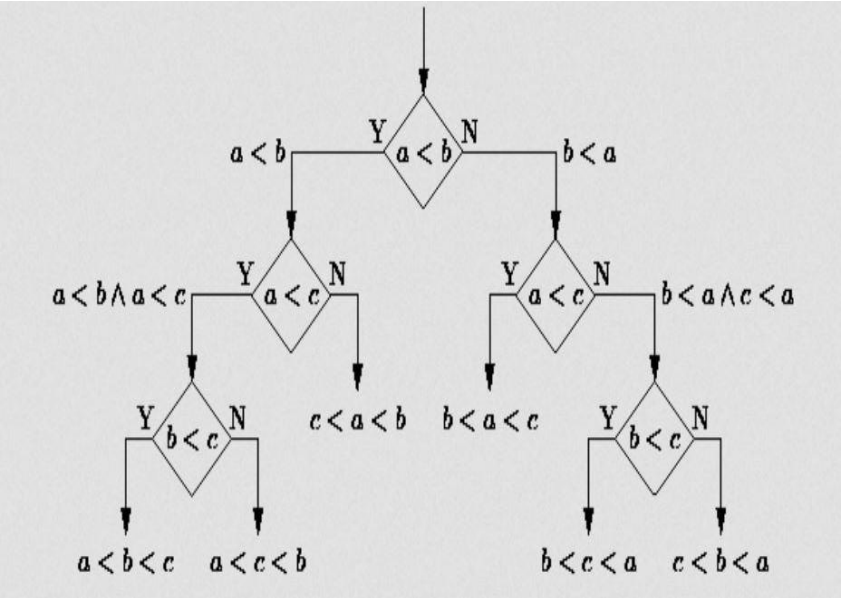


Figure 20: Le fonctionnement d'algorithme l'arbre de décision

Avantages :

-Simple à comprendre et à interpréter.

-Les variables prédictives en entrée peuvent être aussi bien qualitatives que quantitatives.

Inconvénient :

-L'existence d'un risque de sur-apprentissage si l'arbre devient très complexe.

4.6-Les algorithmes de similarité :

Les algorithmes de similarité évaluent la similarité des nœuds à un niveau individuel en fonction des propriétés des nœuds, des nœuds voisins ou des propriétés des relations.

Il existe deux algorithmes de similarité :

- Similitude des nœuds
- Voisins les plus proches approximatifs

4.6.1- Similitude des nœuds :

L'algorithme de similarité de nœud compare un ensemble de nœuds en fonction des nœuds auxquels ils sont connectés. Deux nœuds sont considérés comme similaires s'ils partagent plusieurs des mêmes voisins.

La similarité des nœuds calcule les similarités par paires en se basant sur la métrique « **Jaccard** », connue sous le nom de « **similarité Jaccard** ».

4.6.2- Voisins les plus proches approximatifs :

K-Nearest Neighbors est un algorithme conceptuellement simple mais très puissant, et pour ces raisons, c'est l'un des algorithmes d'apprentissage automatique les plus populaires, et qui peut être utilisé à la fois pour des tâches de régression et de classification.

K-NN examine les étiquettes d'un nombre choisi de points de données entourant un point de données cible, afin de faire une prédiction sur la classe à laquelle appartient ce dernier. [27]

4.6.2.1- Le fonctionnement de l'algorithme KNN :

Un K-NN, il n'y a pas de forme prédéfinie de fonction de mappage.

Considérons la figure suivante :

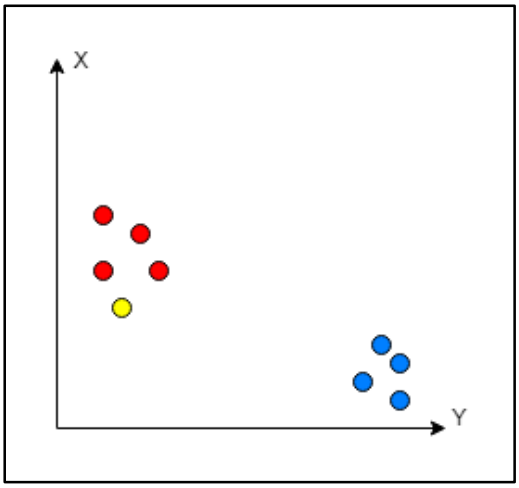


Figure 21: Exemple sur le fonctionnement du K-NN

Disons que nous avons tracé des points de données de notre ensemble d'apprentissage dans un espace de caractéristique bidimensionnel, comme est montré dans la figure ci-dessus, nous avons un total de 8 points de données (4 points rouge appartiennent à la classe 'A' et 4 points bleu appartiennent à la classe 'B'), et le point jaune dans un espace d'entité représente le nouveau point pour lequel une classe doit être prédite. Evidemment, nous disons qu'il appartient à la classe 'A' parce que les voisins les plus proches appartiennent à cette classe

4.6.2.2- les avantages et inconvénients de KNN :

Les avantages :

- Il peut être utilisé à la fois pour des tâches de régression et de classification.
- Il est très précis et simple à utiliser, facile à interpréter, à comprendre et à mettre en œuvre.

L'inconvénient :

- Il stocke la plupart ou la totalité des données. Les ensembles de données volumineux peuvent également faire en sorte que les prédictions prennent beaucoup de temps.

Dans ce sens il existe également des fonctions de similarité comme :

- Similarité de Jaccard
- Similitude du cosinus
- Similitude euclidienne
- Similitude de overlap
- Similitude Pearson

Les fonctions de similarité peuvent être classées en deux groupes. La première est constituée **« catégorique »** mesures qui traitent les tableaux comme des ensembles et calculent la similarité en fonction de l'intersection entre les deux ensembles. La seconde est **« numérique »** mesures qui calculent la similarité en fonction de la proximité des nombres à chaque position les uns par rapport aux autres. [28]

Nous utiliserons les fonctions indiquées dans le tableau ci-dessous

La fonction	Formule	Type	Plage de valeurs
Jaccard	$J(A,B) = \frac{A \cap B}{A \cup B}$	Catégorique	[0,1]
Cosinus	$\cos(ps,pt) = \frac{\sum_i ps(i).pt(i)}{\sqrt{\sum_i ps(i)^2}\ \sqrt{\sum_i pt(i)^2}}$	Numérique	[-1,1]
Euclidienne	$\text{Euclidean (ps, pt)} = \frac{1}{1+\sqrt{\sum_i((\text{ps(i)}-\text{pt(i)})^2}}$	Numérique	[0,1]
Overlap	$\text{O(ps, pt)} = \frac{\lvert ps \cap pt \rvert}{\min(\lvert ps \rvert, \lvert pt \rvert)}$	Catégorique	[0,1]
Pearson	$pearson(ps,pt) = \frac{\sum_i (ps(i) - \overline{ps}).(pt(i) - \overline{ps})}{\sqrt{\sum_i (ps(i) - \overline{ps})^2}\ .\sqrt{\sum_i (pt(i) - \overline{pt})^2}}$	Numérique	[-1,1]

Tableau 3:Les fonctions de similarité

Conclusion

Dans ce chapitre nous avons tenté de présenter de manière simple et complète le concept de quelques méthodes de classification, nous avons donné une vision générale sur les différentes méthodologies de la classification supervisé « les algorithmes du ML » utilisé dans notre étude telle que : les k plus proche voisin (K-NN).
Dans le chapitre suivant, nous présenterons les différents résultats de chaque outil en utilisant la base neo4j et CIDDS et les algorithmes de similarité.

Chapitre 4

Implémentation

Introduction

Notre approche se base sur les graphes de connaissances « Knowledge Graph » et le Machine Learning déjà décrit dans les chapitres précédant, pour extraire des connaissances faire une classification des attaques de la base CIDDS-001 afin d'établir un système de détection d'intrusion basé sur l'analyse du comportement de ces attaques.

Pour cela nous avons montré dans ce chapitre les différentes phases du développement de notre approche, en commençant par une description de la base de données neo4j et CIDDS-001 et les étapes de prétraitement que nous avons fait sur cette dernière.

1-Description de la base Neo4j :

Neo4j est une base de données orientée graphe, libre (sous licence GPLv3) et écrite en Java. Développée par NeoTechnology, les premières lignes de codes datent de l'année 2000 et la version 1.0 est sortie en 2010. [29]

Figure 22:Logo neo4j

Ceci en fait l'une des premières bases de données orientées graphes, mais aussi l'une des plus évoluées et robustes.

Ses principales caractéristiques sont les suivantes :

- **Transaction :** c'est une base de données transactionnelle, respectueuse des principes ACID ;
- **Haute disponibilité :** via la mise en place d'un cluster ;
- **Volumétrie :** stocker et requêter des milliards de nœuds et de relations ;
- **Cypher :** un langage de requête graphe déclaratif, simple et efficace ;

1.1- Neo4j Graph Data Science :

Est une plateforme d'analyse de données et d'apprentissage automatique connectée qui aide à comprendre les connexions dans le Big Data pour répondre aux questions critiques et améliorer les prévisions.

Neo4j pour Graph Data Science comprend les produits suivants :

1-Bibliothèque Neo4j Graph Data Science :

Une boîte à outils avec une structure de données flexible pour l'analyse et une bibliothèque avec cinq variétés d'algorithmes graphiques puissants.

2-Base de donnes de graphesNeo4j :

Une base de données de graphes native hautement évolutive, spécialement conçue pour conserver et protéger les relations.

3-Fleurs de Neo4j :

Un outil de visualisation et d'exploration de graphiques qui permet aux utilisateurs de visualiser les résultats de l'algorithme et de trouver des modèles à l'aide de la recherche sans code.

2- Description du DataSet :

Pour mener à bien notre étude, nous avons choisi le data set **« CIDDS-001 (Coburg Intrusion Detection Data Sets) »**

CIDDS-001 (Coburg Intrusion Detection Data Sets) : est un concept permettant de créer des ensembles de données d'évaluation pour les systèmes de détection d'intrusion réseau basés sur des anomalies.

L'objectif principal de CIDDS-001 est la génération d'ensembles de données personnalisables et à jour. Afin d'atteindre cet objectif, l'idée de base derrière CIDDS-001 est de créer des ensembles de données basés sur des flux étiquetés dans un environnement virtuel à l'aide d'OpenStack. [30]

Le tableau ci-dessus montre un aperçu des attributs dans l'ensemble de données CIDDS-001 :

Contenue des colonnes	Description
Datefirstseen	Start time flow first see
Duration	Durée de flux
Proto	Protocole de transport (ICMP, TCP, UDP)
SrcIPAddr	Addresse IP Source
SrcPt	Port de la Source
DstIPAddr	Addresse IP Destination
DstPt	Port de la Destination
Packets	Nombre de Packet transmis
Bytes	Nombre de Bytes Transmis

Flags	OR concatenation of all TCP Flags
Class	Labele de Classe (normal, attacker, victim)
attackType	Type de l'Attaque (portScan, dos, —)
attackID	Identifiant d'attaque. Tous les flux appartenant à la même attaque portent le même identifiant d'attaque.
attackDescription	Fournit des informations supplémentaires sur les paramètres d'attaque définis

Tableau 4:Description des champs du DataSet

3- Cypher :

Cypher est un langage déclaratif permettant de requêter et mettre à jour le graphe. Inspiré du SQL, on y retrouve beaucoup de concepts familiers, comme les clauses WHERE, ORDER BY, SKYP, LIMIT…

Son objectif est de permettre à l'utilisateur de définir des motifs, qui seront par la suite recherchés dans tout le graphe.

4- Implémentation :

Une fois lancer le neo4j s'affiche la fenêtre suivante :

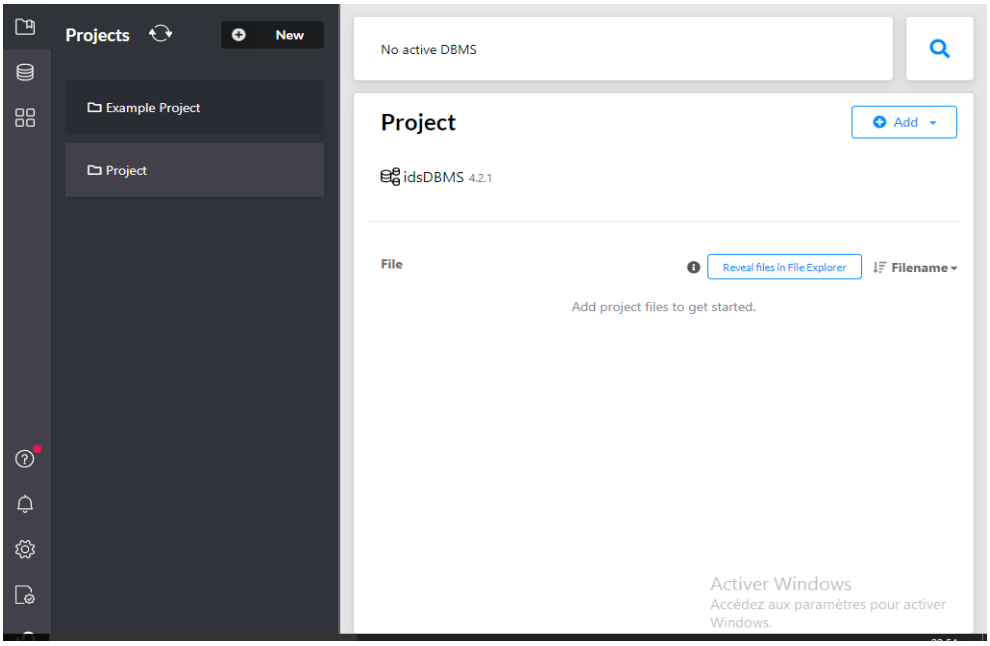


Figure 23:La première fenêtre sur neo4j

Nous créerons un nouveau projet comme suit :

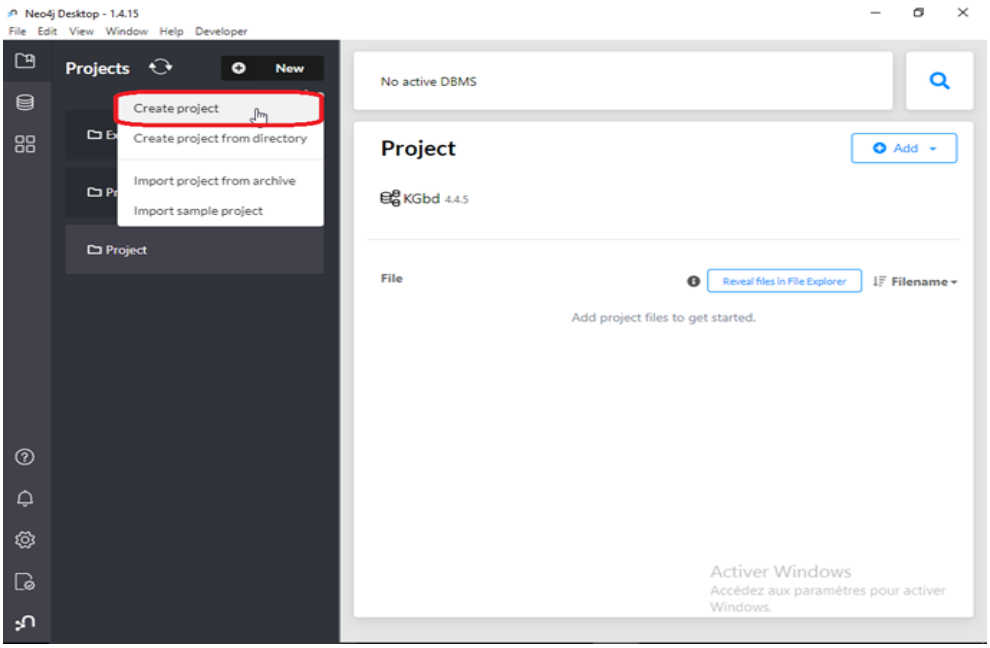


Figure 24:Création du projet

Puis créer une base de données locale

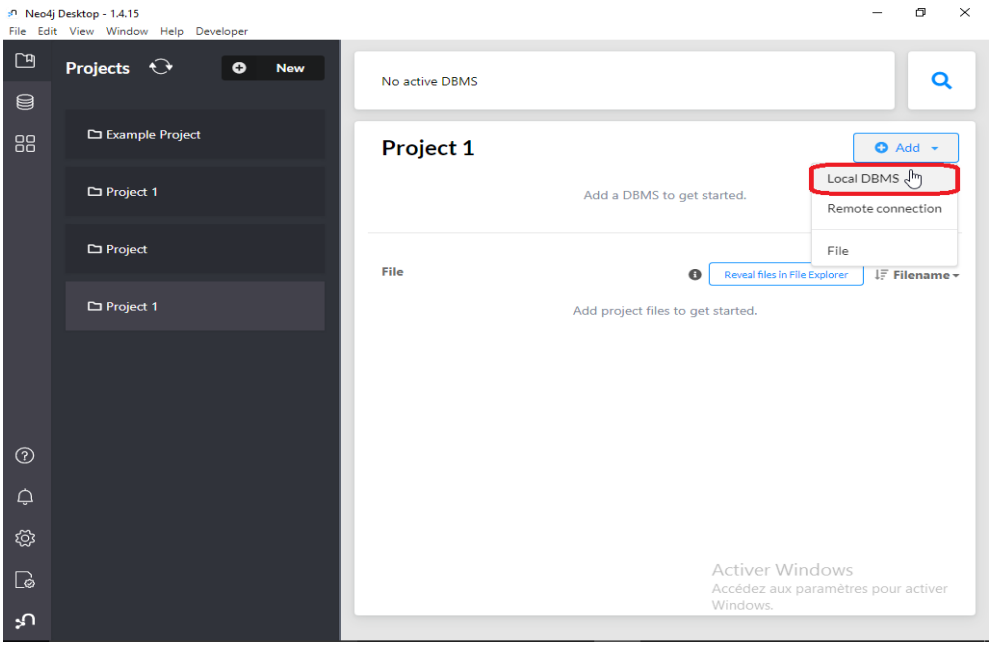


Figure 25: Création du DBMS

Ensuite l'importation du fichier csv

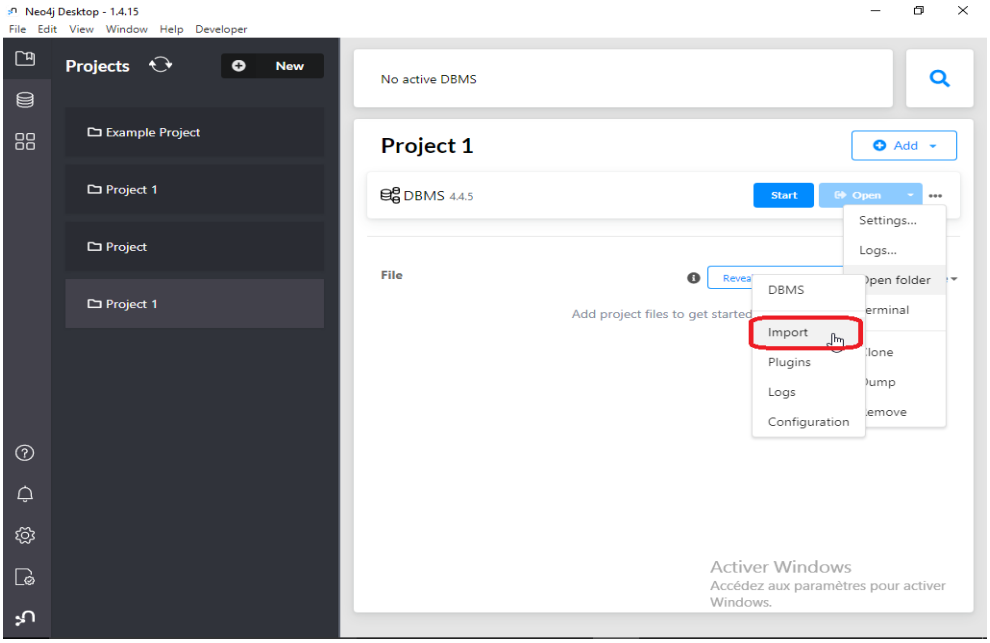


Figure 26: Importation de fichier CSV

Nous ouvrirons notre projet

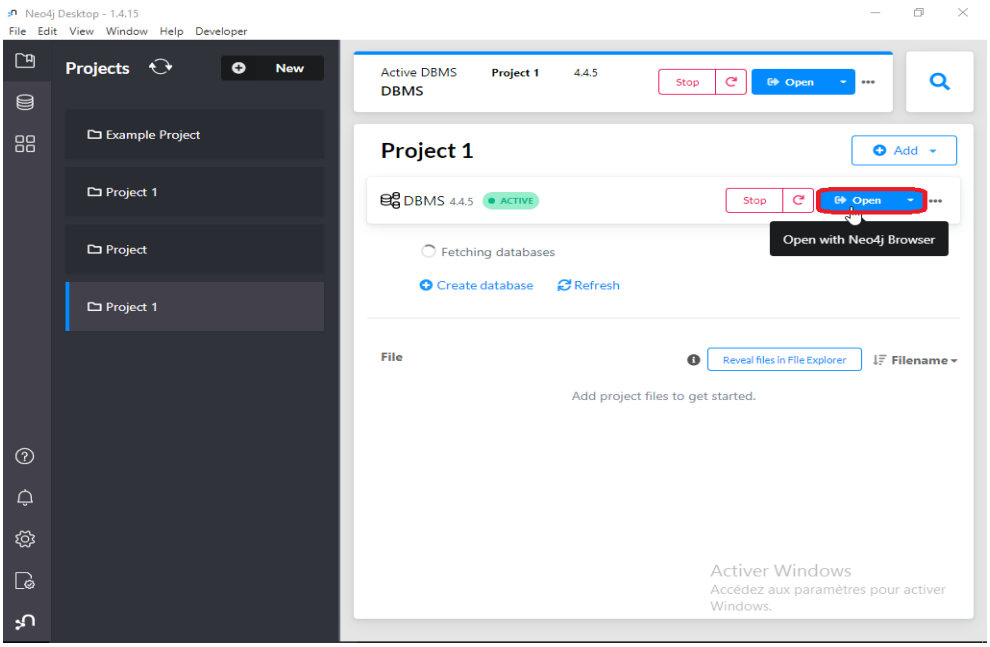


Figure 27: L'ouverture de projet

Etape 1 : le chargement du data set avec la commande cypher comme suit :

Syntaxe :

```
load csv with headers from 'file:///CIDDS.csv' as row
return (row) limit 25 ;
```

Figure 28: Chargement de DataSet

Etape 2 : la création des contraintes :

Syntaxe :

```
CREATE CONSTRAINT NOT EXISTS attackID ON (a:attackType) ASSERT a.name IS UNIQUE
```

Figure 29: Création des contraintes

Etape3 : la création des nœuds se fait comme suit :

```
load csv with headers from 'file:///CIDDS.csv' as row
MERGE (a:attackType {name: row.attackType})
SET a.attacks = row.attackType;
```

Figure 30: Création des nœuds

Etape4 : construire des relations entre les nœuds :

```
load csv with headers from 'file:///CIDDS.csv' as row
MATCH (a:attackType {name: row.attackType})
MATCH (aid:attackID {name: row.attackID})
MERGE (aid)-[:IS_]->(a)
```

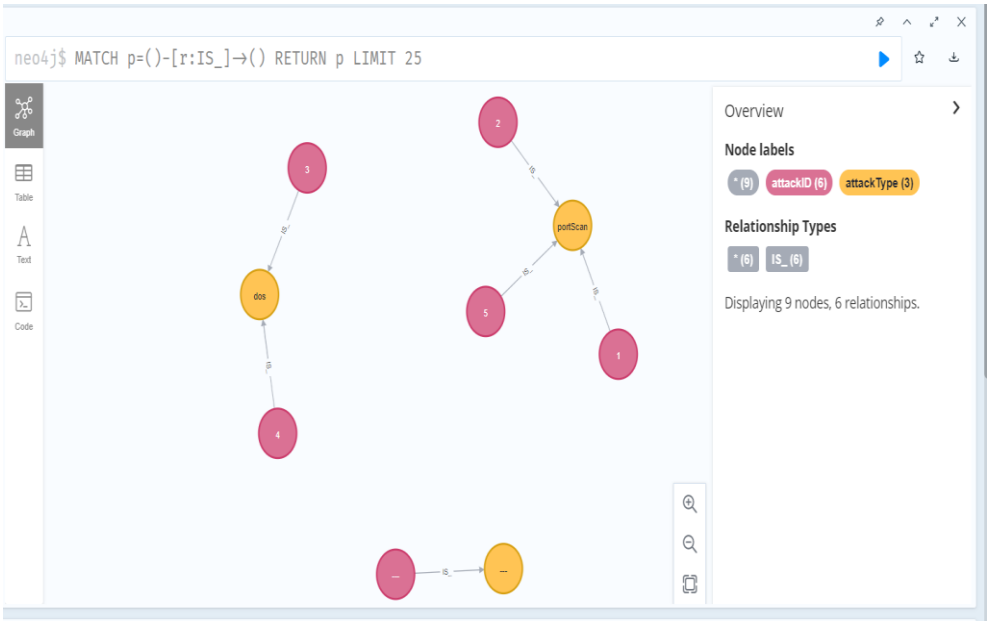


Figure 31: Création des relations

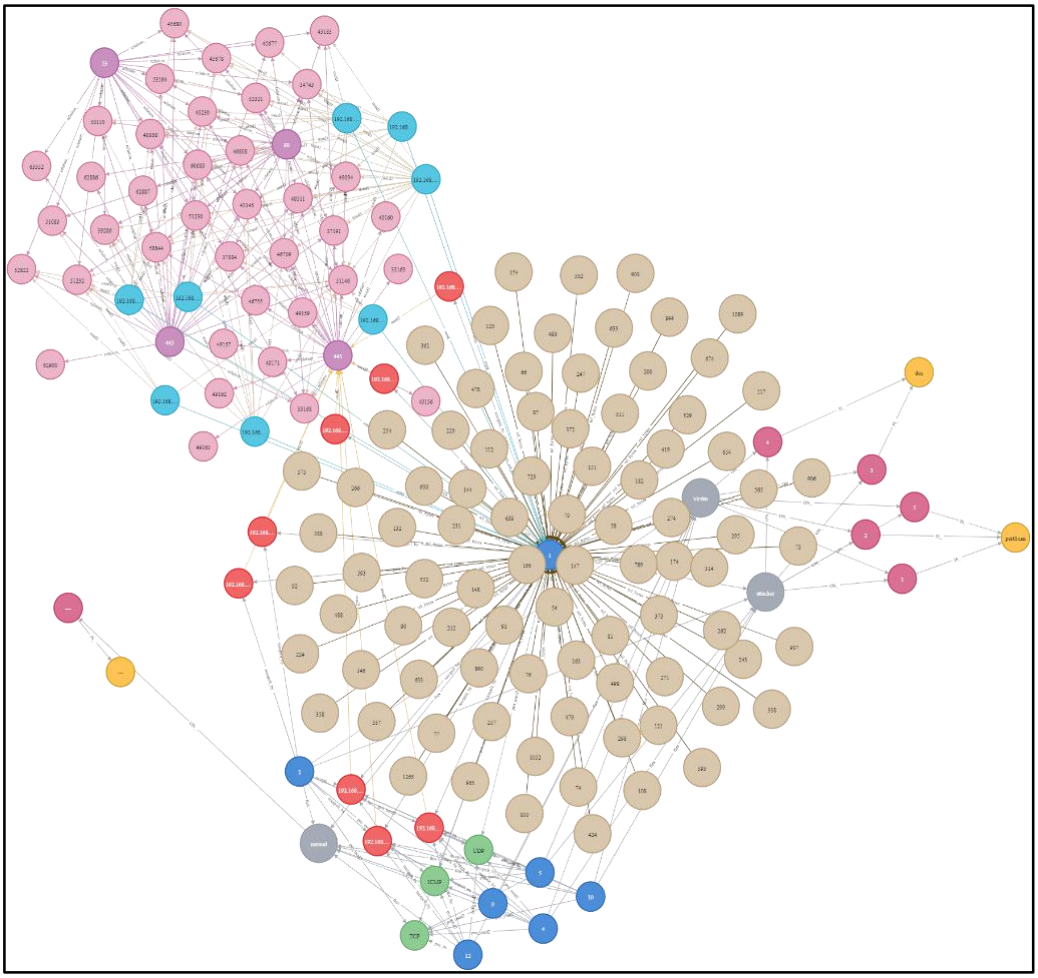


Figure 32:Visualisation de Graphe.

Pour mieux comprendre le contenu du DataSet nous utiliserons les algorithmes de similarité déjà décrit dans les chapitres précédents.

La bibliothèque Neo4j GDS inclut les algorithmes de similarité suivants :

- Similitude de nœud
- K-Voisins les plus proches

1. Similitude de nœud :

La syntaxe générale de l'algorithme implique de référencer un graphe nommé précédemment chargé. De plus, différents modes d'exécution sont fournis :

Etape1 : Projection de Graph data science :

Syntaxe :

```
CALL gds.graph.project(
   'GraphDS',
   {
      Packets:{       label: 'Packets'},
     SrcIPAddr:{    label: ' SrcIPAddr '} ,
     DstIPAddr: {   label: ' DstIPAddr '},
      Proto: {          label: ' Proto'},
     attackID:{       label: 'attackID'}
    '*'
);
```

Ce graphe a cinq ensembles de nœuds: les nœuds Packets, les nœuds de l'Addresse IP source, les nœuds de l'Addresse IP destination, les nœuds de Protocole et les nœuds Identifiant de l'attaque. Les ensembles de nœuds sont connectés via des relations, nous voulons utiliser l'algorithme NodeSimilarity pour comparer les Packets en fonction de leur adresse IP source et destination aussi en fonction de leur protocole et l'identifiant de l'attaque.

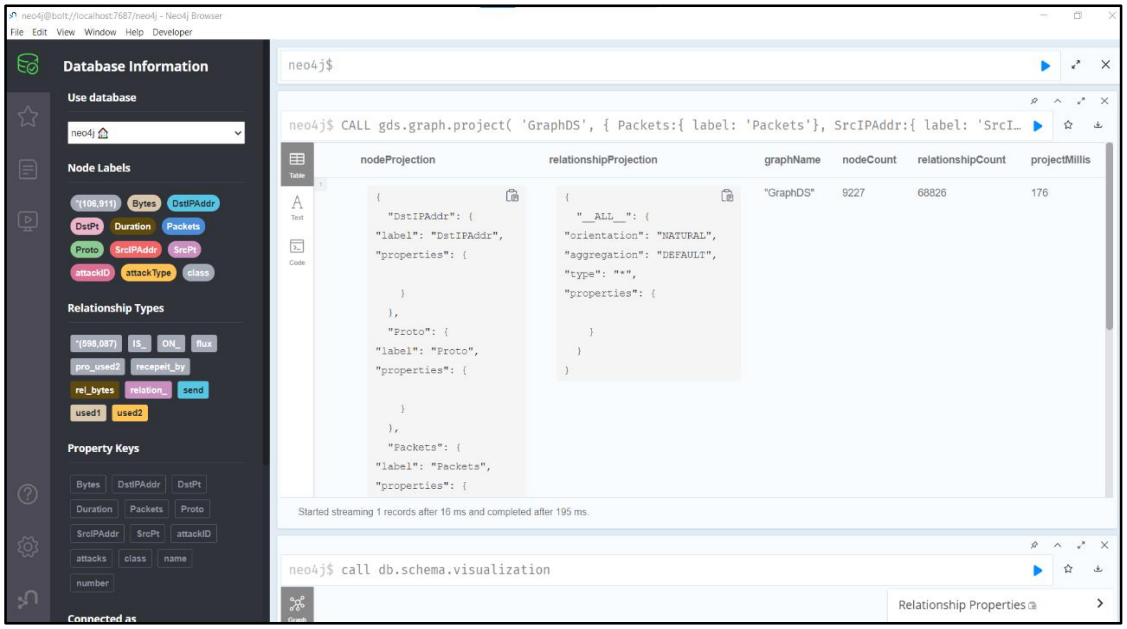


Figure 33 : Projection de GraphDS

Etape2 : L'exécution de fonction estimate pour estimer le cout d'exécution de l'algorithme (similitude des nœuds). Cela peut être fait avec n'importe quel mode d'exécution. Nous utiliserons le **write** mode dans cette étape.

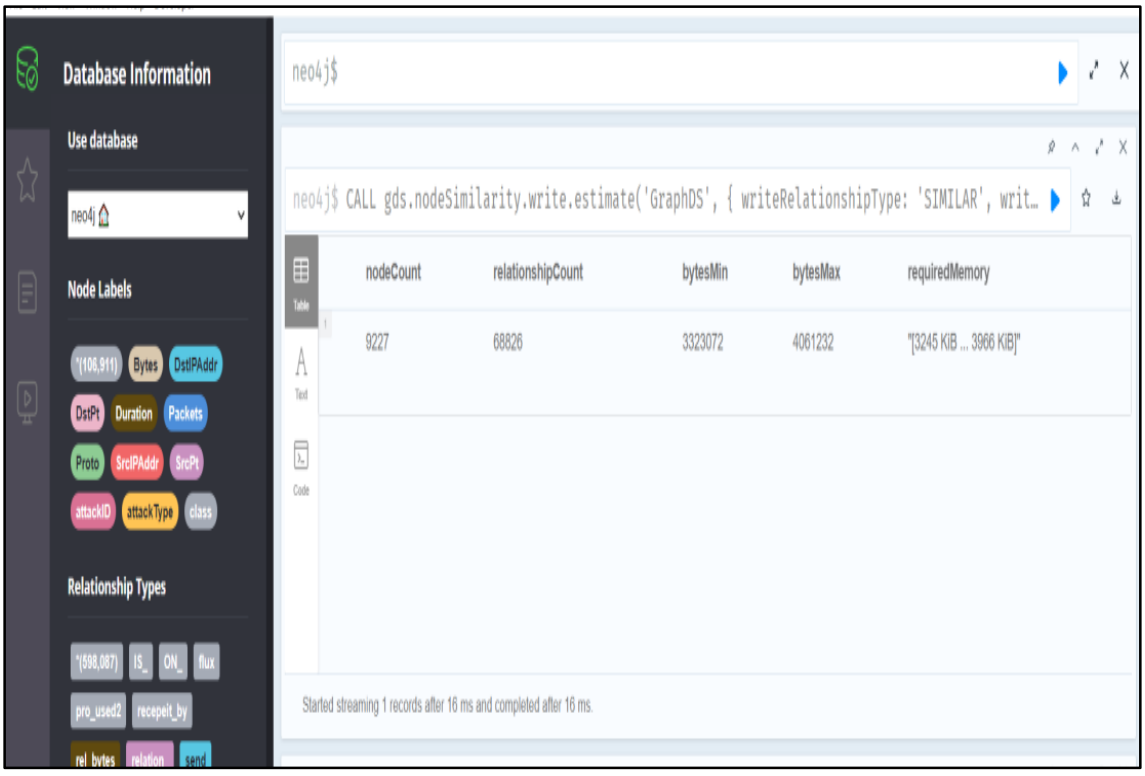


Figure 34:mode estimate

1 -Mode Flux :

En **Stream** mode exécution, l'algorithme renvoie le score de similarité pour chaque relation

```
CALL gds.nodeSimilarity.stream('GraphDS')
YIELD node1, node2, similarity
RETURN gds.util.asNode(node1).name AS colonne1, gds.util.asNode(node2).name AS colonne2, similarity
ORDER BY similarity DESCENDING, colonne1, colonne2
```

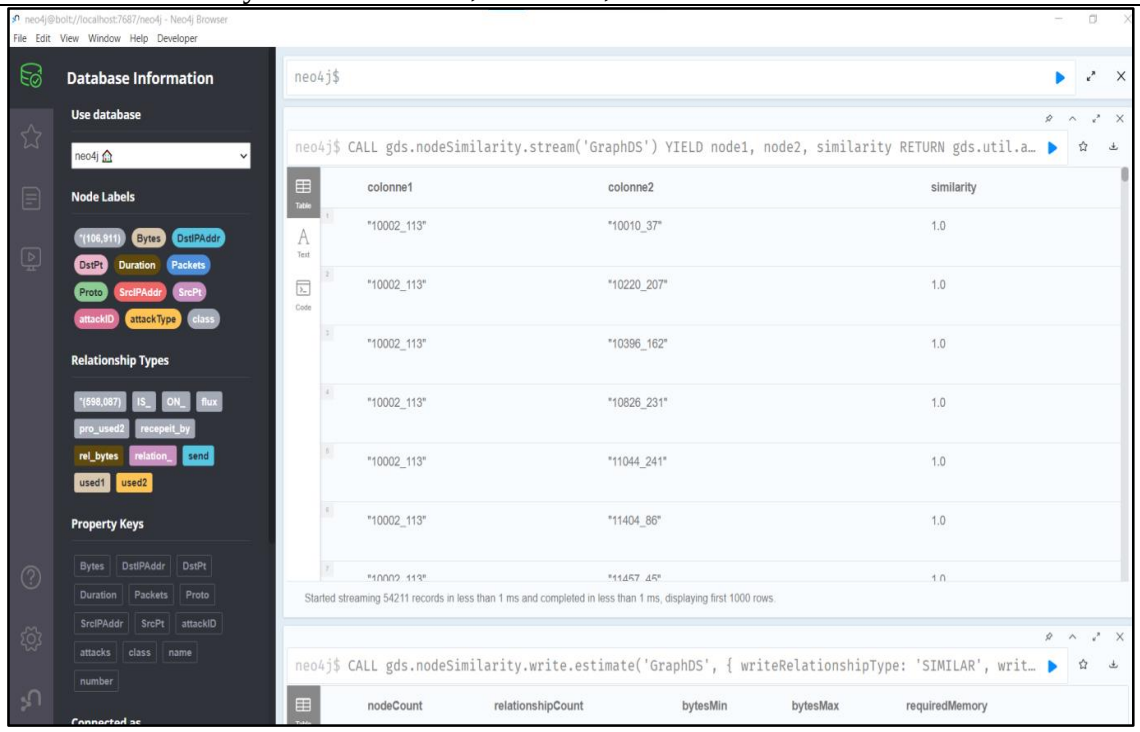


Figure 35:Mode flux

2 –Mode Stats :

L'algorithme renvoie une seule ligne contenant un résumé du résultat de l'algorithme

```
CALL gds.nodeSimilarity.stats('GraphDS')
YIELD nodesCompared, similarityPairs
```

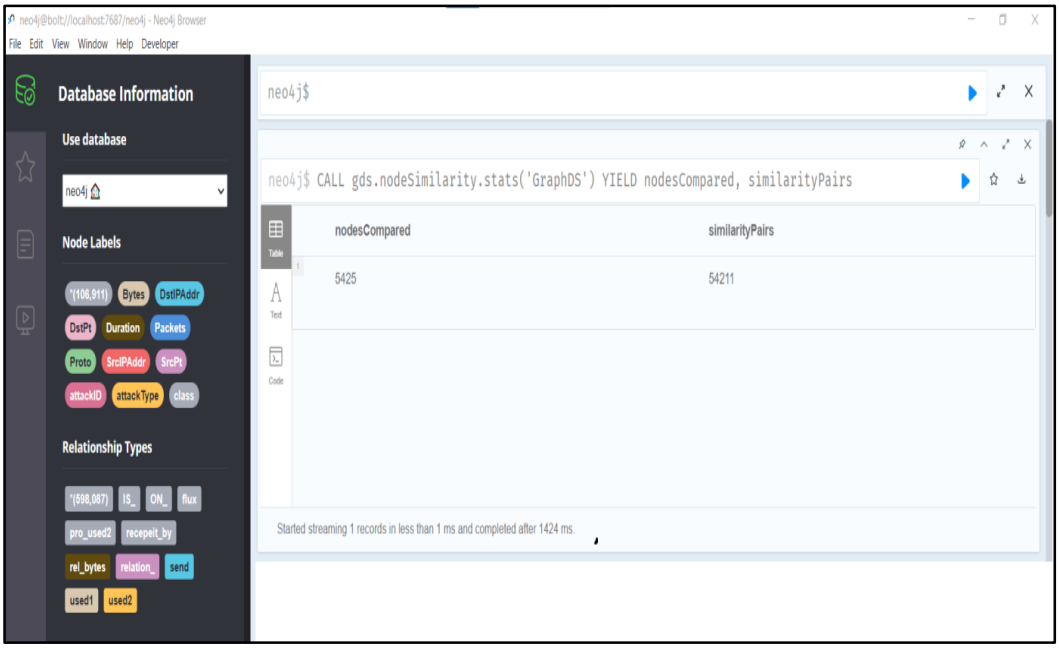


Figure 36:Mode stats

Nom	Taper	La description
nodesCompared	Entier	Le nombre de nœuds pour lequel la similarité a été calculée
similarityPairs	Entier	Le nombre de similitudes dans le résultat

Tableau 5: description du mode stats

3 –Mode Mutate:

Mettre à jour le graphe nommé avec une nouvelle propriété de relation contenant le score de similarité pour cette relation. Le nom de la nouvelle propriété est spécifié à l'aide du paramètre de configuration obligatoire **mutateProperty.**

```
CALL gds.nodeSimilarity.mutate('GraphDS', {
    mutateRelationshipType: 'SIMILAR_Test',
    mutateProperty: 'score_test'
})
```

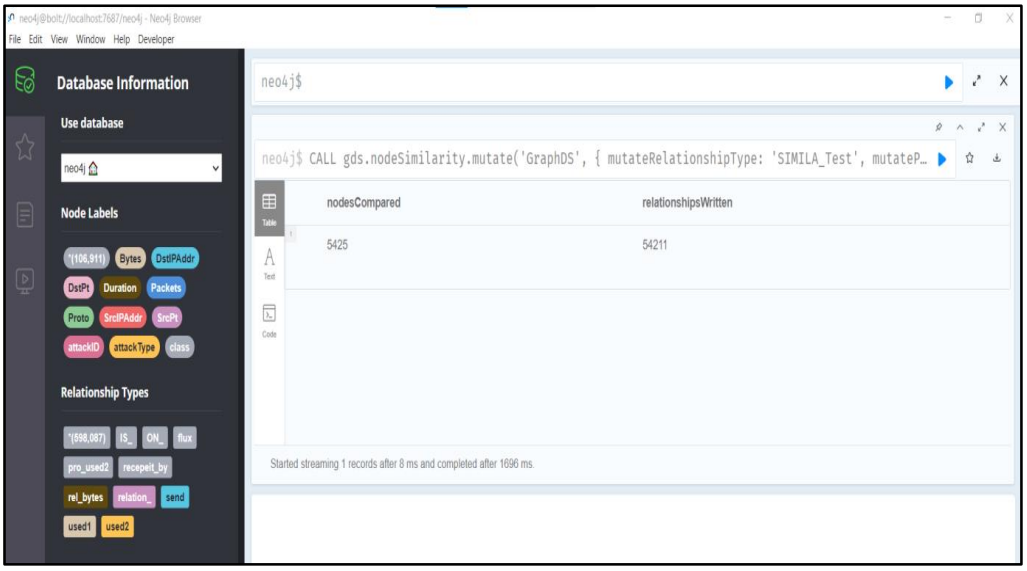


Figure 37:Mode mutate.

4- Mode Write :

Le **write** mode d'exécution de chaque paire de nœuds crée une relation avec leur score de similarité en tant que propriété de la base de données Neo4j. Le type de la nouvelle relation est spécifié à l'aide du paramètre de configuration obligatoire **writeRelationshipType.**

```
CALL gds.nodeSimilarity.write('myGraphe', {
    writeRelationshipType: 'SIMILAR_test',
    writeProperty: 'score_test'
})
YIELD nodesCompared, relationshipsWritten
```

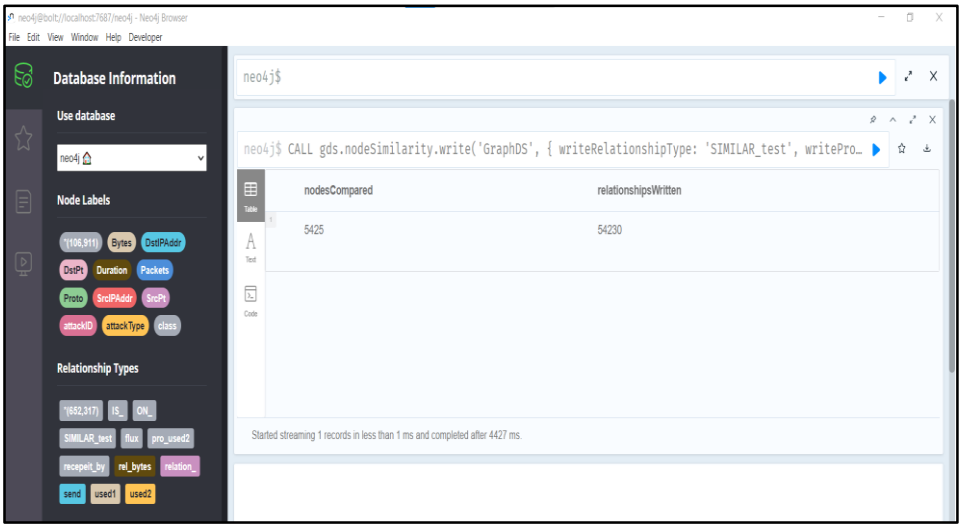


Figure 38:Mode write

2-Travail et résultats de l'algorithme K-NN :

Projection du graphe Data science nommé Packets :

```
CALL gds.graph.project(
   'Packets',
     {Packets:
     {properties:['Packets']}

       },'*'

)
```

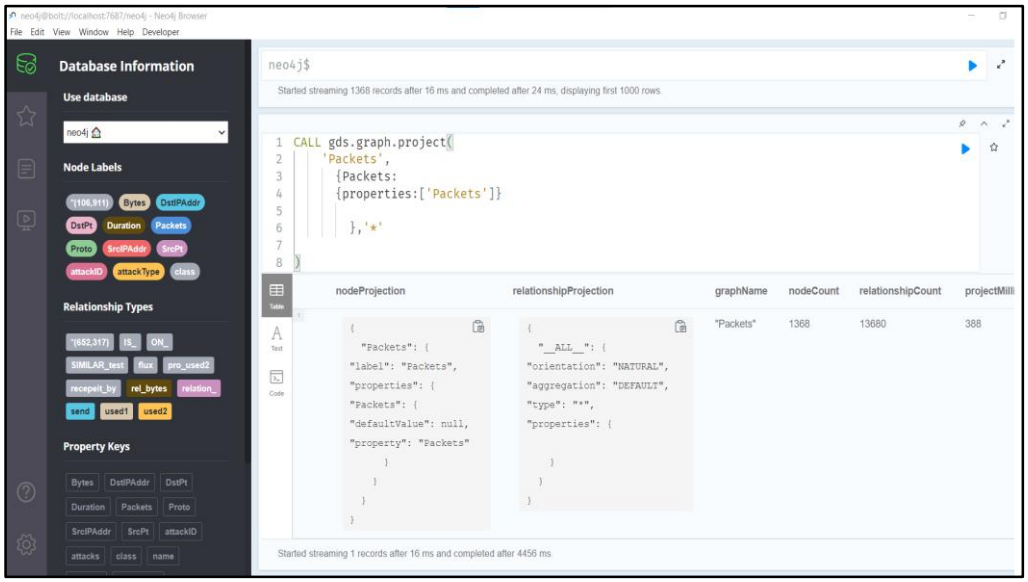


Figure 39:Projection de Graphe Packets

Mode Write :

```
CALL gds.knn.write('Packets', {
   writeRelationshipType: 'SIMILAR_KNN',
   writeProperty: 'score_KNN',
   topK: 10,
   randomSeed: 42,
   concurrency: 1,
   nodeProperties: 'Packets'
})
YIELD nodesCompared, relationshipsWritten

// Explore the similarity results
match (PK1:Packets)-[:SIMILAR_KNN]->(PK2)
RETURN PK1, PK2
LIMIT 500
```

Le **write** mode d'exécution a un effet important pour chaque paire de nœuds, nous créons une relation avec le score de similarité en tant que propriété de la base de données Neo4j. Le type de la nouvelle relation est spécifié à l'aide du paramètre de configuration obligatoire **writeRelationshipType** Chaque nouvelle relation stocke le score de similarité entre les deux nœuds qu'elle représente. La clé de propriété de relation est définie à l'aide du paramètre de configuration obligatoire **writeProperty**.

La figure suivante résume l'exécution l'algorithme **knn** pour les Packets :

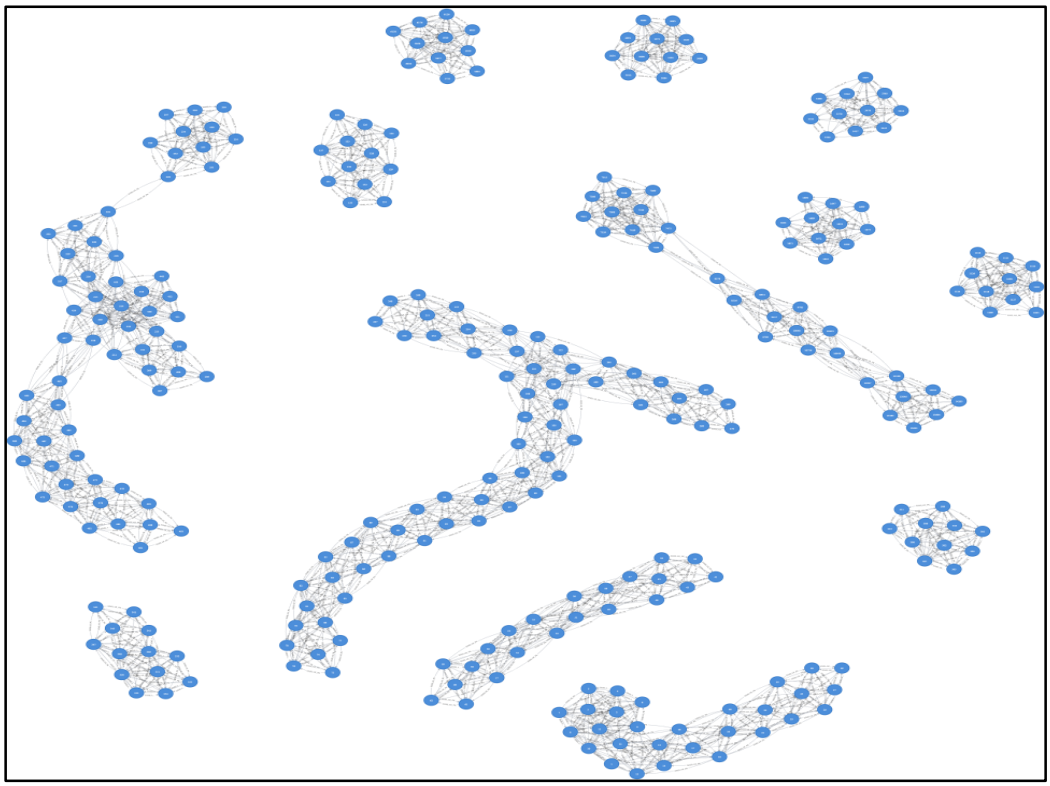


Figure 40:Resultat finale du KNN sur le graphe Packets

Conclusion

Nous avons abordé dans ce dernier chapitre la base de données CIDDS qui a été utilisée pour l'apprentissage et test du modèle de détection d'intrusions généré basé sur les algorithmes de similarité.

Nous avons aussi expliqué le fonctionnement de l'application neo4j pour mettre en œuvre le modèle de classification. Ensuite, nous présentons les résultats obtenus après la combinaison des graphes de connaissance avec les algorithmes de similarité (la similitude de nœuds et K-NN) comme technique de classification.

Les résultats ont également montré l'importance des algorithmes de similarité dans l'augmentation des performances de l'approche générée.

Conclusion générale

Les systèmes de détection d'intrusion faites partie des activités de surveillance des réseaux visent à protéger les fonctionnalités et l'intégrité des données. Généralement, les IDS sont basés sur les techniques de machine Learning dans la conception de leur modèle. Ces IDS sont généralement basé sur les modèles des machines Learning qui nous ramène à des résultats d'une façon générale.

Ce travail a pour objectif de combiner les KG et les techniques de ML afin d'extraire des nouvelles connaissances et de classifier les attaques. Cela a été fait par offrir un graphe de connaissances plus complet pour la gestion des données du data-set utilisé CIDDS-001 pour la détection d'intrusion basé sur le réseau, en rendant les données plus intelligentes.

En premier lieu, on a commencé par la description des systèmes de détection d'intrusion par la présentation de ces différentes architectures, ces types et ces méthodes de détection. Ensuite, nous avons touché les techniques de machine Learning qui sont une étape essentielle pour la construction des modèles d'IDS. Après un chapitre qui décrit notre sujet, c'est les graphes de connaissances qui sont une innovation a la représentation des connaissances. A la fin, nous avons présenté notre implémentation par la construction du graphe de connaissance par sa création des nœuds et les relations. Après sa création, un ensemble des algorithmes ont été appliqués afin de découvrir des nouvelles connaissances. Les résultats obtenus sont satisfaisants dont notre objectif est atteint par l'organisation des données sous forme des graphes de connaissances simples et lisibles

Nous concluons que l'objectif de ce modeste travail a une grande capacité à fournir des informations sémantiquement structurées et que des avancées importantes dans l'application de cette capacité détection des attaques spécifiques ont été réalisées. Une classification a été affichée par la combinaison de méthodes de machine Learning et du graphe de connaissances, qui fournit intuitivement une base raisonnable. De plus, les résultats expérimentaux montrent que le cadre proposé atteint de meilleures visualisations sémantiques pour des nouvelles connaissances.

Bibliographie :

[1] [En ligne]. Available: https://www.lemagit.fr/definition/Systeme-de-detection-dintrusions. [Accès le Janvier 2022].

[2] [En ligne]. Available: T. Simon, «Détection d'intrus dans les reseaux a l'aide d'agents mobile,» Montréal, 2006.. [Accès le 2 Fevrier 2022].

[3] [En ligne]. Available: https://www.checkpoint.com/cyber-hub/network-security/what-is-an-intrusion-detection-system-ids/. [Accès le Janvier 2022].

[4] [En ligne]. Available: https://logicalread.com/intrusion-detection-system/#.YqnSb6jMJxA. [Accès le Janvier 2022].

[5] [En ligne]. Available: K.Boudaoud, «Systéme multi-agents pour la détection d'intrusions,» 24 10 2000. [En ligne].. [Accès le 1 Fevrier 2022].

[6] [En ligne]. Available: D. Riquet, «Une architecture de détection d'intrusions reseau distribuéebasée sur un langage dédié,» France, 2015.. [Accès le fevrier 2022].

[7] [En ligne]. Available: H. Belkadi, «Détaction d'intrusion dans le cloud computing,» Bejaia, 2016.. [Accès le 2022].

[8] [En ligne]. Available: «IDS - Systèmes de détection d'intrusion,» [En ligne]. Available: https://web.maths.unsw.edu.au/~lafaye/CCM/detection/ids.htm.. [Accès le 18 Decembre 2021].

[9] [En ligne]. Available: T. Simon, «Détection d'intrus dans les reseaux a l'aide d'agents mobile,» Montréal, 2006.. [Accès le 25 Decembre 2021].

[10] [En ligne]. Available: https://www.maxicours.com/se/cours/graphes-definitions-proprietes/. [Accès le 18 Avril 2022].

[11] [En ligne]. Available: https://medium.com/analytics-vidhya/introduction-to-knowledge-graphs-and-their-applications-fb5b12da2a8b. [Accès le 19 Mars 2022].

[12] [En ligne]. Available: https://www.smalsresearch.be/les-graphes-de-connaissance-incontournable-pour-lintelligence-artificielle-2/. [Accès le Mars 2022].

[13] [En ligne]. Available: https://linkeddigitalfuture.ca/fr/2019/09/12/graphe-de-connaissances/. [Accès le 22 Avril 2022].

[14] [En ligne]. Available: https://linkeddigitalfuture.ca/fr/2019/09/12/graphe-de-connaissances/. [Accès le Avril 2022].

[15] [En ligne]. Available: https://medium.com/analytics-vidhya/introduction-to-knowledge-graphs-and-their-applications-fb5b12da2a8b. [Accès le Avril 2022].

[16] [En ligne]. Available: https://medium.com/analytics-vidhya/introduction-to-knowledge-graphs-and-their-applications-fb5b12da2a8b. [Accès le Mars 2022].

[17] [En ligne]. Available: https://www.smalsresearch.be/les-graphes-de-connaissance-incontournable-pour-lintelligence-artificielle-2/. [Accès le Janvier 2022].

[18] [En ligne]. Available: https://www.codetd.com/fr/article/12670772. [Accès le 22 Avril 2022].

[19] Jesús Barrasa, Amy E. Hodler, and Jim Webber, «Knowledge Graphs Data in Context for Responsive Businesses,» the United States of America.

[20] [En ligne]. Available: https://www.smalsresearch.be/les-graphes-de-connaissance-quelques-applications/. [Accès le Avril 2022].

[21] [En ligne]. Available: https://www.europarl.europa.eu/news/fr/headlines/society/20200827STO85804/intelligence-artificielle-definition-et-utilisation. [Accès le Mai 2022].

[22] [En ligne]. Available: https://www.journaldunet.com/solutions/dsi/1153939-machine-learning-les-dessous-techniques-d-une-revolution-technologique/. [Accès le Mai 2022].

[23] A. L. Pauline Le Badezet, «Classification Exemple : Enquête d'opinion sur les OGM».

[24] [En ligne]. Available: https://www.linkedin.com/pulse/les-types-de-machine-learning-william-simetin-grenon/. [Accès le 10 Mai 2022].

[25] [En ligne]. Available: https://mrmint.fr/9-algorithmes-de-machine-learning-que-chaque-data-scientist-doit-connaitre. [Accès le Mai 2022].

[26] [En ligne]. Available: https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm. [Accès le 7 Mai 2022].

[27] [En ligne]. Available: https://www.unite.ai/what-is-k-nearest-neighbors/. [Accès le Mai 2022].

[28] [En ligne]. Available: https://neo4j.com/docs/graph-data-science/current/algorithms/node-similarity/. [Accès le Mai 2022].

[29] [En ligne]. Available: https://logisima.developpez.com/tutoriel/nosql/neo4j/introduction-neo4j/.. [Accès le Avril 2022].

[30] [En ligne]. Available: https://www.hs-coburg.de/forschung/forschungsprojekte-oeffentlich/informationstechnologie/cidds-coburg-intrusion-detection-data-sets.html. [Accès le Mai 2022].