ANALYSE DES PERFORMANCES DES CLASSIFICATEURS DE GAIN MACHINE L POUR LA DÉTECTION D'INTRUSION À L'AIDE DE L'ENSEMBLE DE DONNÉES UNSW-NB15

Geeta Kocher1 et Gulshan Kumar2

[1]Chercheur, Département des sciences computationnelles,
MRSPTU, Bathinda, Pendjab, Inde
[2]Professeur associé, Département des applications informatiques, SBSSTC, Ferozpur, Pendjab,
Inde

ABSTRAIT

Avec l'avancement de la technologie Internet, le nombre de menaces augmente également de manière exponentielle. Pour réduire l'impact de ces menaces, les chercheurs ont proposé de nombreuses solutions de détection d'intrusion. Dans la littérature, divers classificateurs d'apprentissage automatique sont formés sur des ensembles de données plus anciens pour la détection d'intrusion, ce qui limite leur précision de détection. Il est donc nécessaire de former les classificateurs d'apprentissage automatique sur le dernier ensemble de données. Dans cet article, UNSW-NB15, le dernier ensemble de données est utilisé pour former des classificateurs d'apprentissage automatique. Sur la base d'une analyse théorique, la taxonomie est proposée en termes d'apprenants paresseux et avides. À partir de cette taxonomie proposée, les classificateurs K- Nearest Neighbors (KNN), Stochastic Gradient Descent (SGD), Decision Tree (DT), Random Forest (RF), Logistic Regression (LR) et Naïve Bayes (NB) sont sélectionnés pour la formation. La performance de ces classificateurs est testée en termes d'exactitude, d'erreur quadratique moyenne (MSE), de précision, de rappel, de score F1, de taux de vrais positifs (TPR) et de taux de faux positifs (FPR) sur l'ensemble de données UNSW-NB15 et une analyse comparative de ces classificateurs d'apprentissage automatique est effectuée. Les résultats expérimentaux montrent que le classificateur RF surpasse les autres classificateurs.

MOTS-CLÉS

Système de détection d'intrusion, forêt aléatoire, KNN, UNSW-NB15, classificateurs d'apprentissage automatique

1. INTRODUCTION

De nos jours, sécuriser les données confidentielles de l'œil des attaquants devient une tâche cruciale et difficile. Les méthodes traditionnelles comme le pare-feu et l'antivirus ne sont pas suffisantes pour faire face à tous les types d'attaques. Il y a donc un besoin de sécurité supplémentaire avec les méthodes traditionnelles. Le système de détection d'intrusion (IDS) joue un rôle important à cet égard. Il garde soigneusement une trace des données de trafic réseau et distingue si les données sont normales ou d'attaque.

Un IDS est utilisé pour surveiller le trafic réseau afin de détecter les activités malveillantes. Il peut facilement détecter les attaques contournées par le pare-feu. Il surveille en permanence le réseau, trouve les parties vulnérables du réseau et communique à l'administrateur les intrusions [1]. Il peut être séparé en deux classes: la détection d'anomalies et la détection d'abus. La détection des abus fonctionne avec des modèles d'attaques connues préparés à l'avance, également appelés signatures [2]. Il a une grande précision et de faibles taux de fausses alarmes (FAR), mais incapable de détecter de nouvelles attaques [3]. L'une des solutions pour résoudre ce

problème est de mettre régulièrement à jour la base de données, ce qui n'est pas faisable et un processus coûteux. Ainsi, les techniques de détection des anomalies ont vu le jour. La détection d'anomalies traite du profilage de l'utilisateur

David C. Wyld et al. (Eds): SIGI, CSTY, AI - 2020
p. 31 à 40, 2020. CS & IT - PCSC 2020 DOI: 10.5121/csit.2020.102004

comportement [4]. Dans cette approche, un certain modèle d'activité normale de l'utilisateur est défini, et tout écart par rapport à ce modèle est connu comme anormal. La figure 1 montre le diagramme de l'IDS.

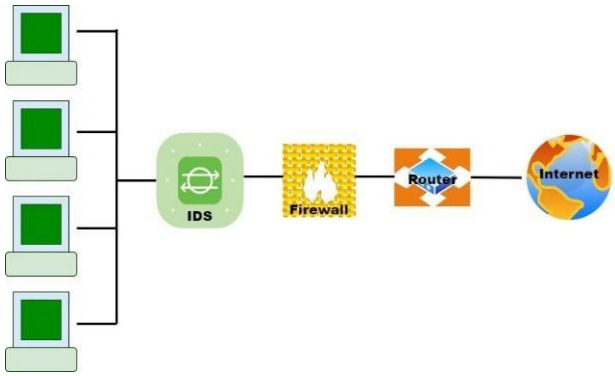

Fig 1: Système de détection d'intrusion

Dans la littérature, différents types de classificateurs d'apprentissage automatique (ML) sont utilisés pour la détection d'intrusion. D'après la littérature, on constate que les travaux d'analyse comparative des classificateurs de ML sont limités. Par conséquent, le motif de cet article est de trouver une comparaison des performances de plusieurs classificateurs ML en utilisant un ensemble de données récent pour la détection d'intrusion. La structure du document est divisée en sept sections. La section 2 donne un bref aperçu de la littérature liée à ce travail de recherche. Dans la section 3, la taxonomie des classificateurs est discutée. Une brève introduction sur l'ensemble de données utilisé pour les travaux expérimentaux est décrite à la section 4. La section 5 présente une méthodologie de prétraitement de l'ensemble de données. Les travaux expérimentaux sont présentés à la section 6 et la section 7 donne la conclusion et la portée future.

2. TRAVAUX CONNEXES

Cette section fournit l'étude de la littérature sur les classificateurs ML. L'objectif principal de cette section est de donner un aperçu des travaux de recherche effectués dans le domaine de la détection d'intrusion. Il ressort de la littérature que les chercheurs ont consacré beaucoup d'efforts aux classificateurs ML et certaines de leurs contributions sont décrites ci-dessous:

Narudin et. al. (2014) [5] a décrit une évaluation utilisant des classificateurs ML, à savoir RF, J-48, Multilayer Perceptron (MLP), NB et KNN pour détecter les logiciels malveillants mobiles à l'aide de deux ensembles de données, à savoir MalGenome et Private. L'outil Weka a été utilisé pour l'évaluation. Les paramètres de rendement, à savoir TPR, FPR, précision, rappel et f-mesure, ont été utilisés pour valider le rendement des classificateurs ML. La précision obtenue par RF Classifier est de 99,99% lors de travaux expérimentaux sur l'ensemble de données MalGenome. L'auteur a suggéré d'améliorer les résultats en utilisant des fonctionnalités sélectionnées pour des travaux futurs.

Belavagi & Muniyal, (2016) [6] ont conçu un système de détection d'intrusion réseau (NIDS) avec les différents classificateurs d'apprentissage automatique supervisés. L'ensemble de données NSL-KDD a été utilisé pour vérifier les performances de divers classificateurs. Le résultat montre que le classificateur RF surpasse les autres classificateurs. Il en résulte un FPR le plus bas et un TPR le plus élevé et la précision obtenue est de 99%. Mais encore, il y a un besoin de classificateurs qui peuvent être utilisés pour la classification multiclasse.

Ashfaq et al, (2017) [7] ont décrit une approche d'apprentissage semi-supervisé (SSL) basée sur le nouveau flou. Afin d'améliorer les performances du classificateur, il utilise des échantillons non étiquetés ainsi qu'un algorithme d'apprentissage supervisé. L'ensemble de données NSL-KDD a été utilisé pour évaluer ce modèle.

La limite de ce modèle était que ses performances n'étaient étudiées que pour la tâche de classification binaire.

Yaseen et al, (2017) [8] ont décrit un modèle hybride de détection d'intrusion à plusieurs niveaux utilisant Support Vector Machine (SVM) et Extreme Learning Machine (EVM). L'évaluation a été effectuée sur l'ensemble de données KDD 99. La précision obtenue était de 95,75% et le temps de formation était plus court dans ce modèle proposé. Cette technique n'est meilleure que pour les attaques connues et pour les nouvelles attaques, des classificateurs efficaces sont nécessaires.

Aljumah, (2017) [9] a décrit un algorithme entraîné pour détecter les attaques DDoS qui était basé sur un réseau de neurones artificiels (ANN). ANN montre une précision de 92% lorsqu'il a été formé avec des ensembles de données plus anciens et lorsque le système est entraîné avec des ensembles de données mis à jour, la précision obtenue était de 98%. La précision du modèle ANN dépend de l'ensemble de données. Il est donc nécessaire de disposer d'un ensemble de données à jour et équilibré.

Roshan et al., (2018) [10] ont discuté d'une conception adaptative de l'IDS basée sur les machines d'apprentissage extrême (ELM). L'ensemble de données NSL-KDD a été appliqué à l'évaluation. Il a été constaté qu'il peut détecter de nouvelles attaques ainsi que des attaques connues avec un taux de détection acceptable et des faux positifs.

Ali et al., (2018) [11] ont proposé un classificateur PSO-FLN pour la détection des intrusions. L'ensemble de données de référence KDD99 a été utilisé pour valider les résultats. Le PSO-FLN a surpassé les classificateurs ELM et FLN en termes de précision. Mais pour certaines classes comme R2L, il ne montre pas de résultats précis. L'étude de la littérature permet de conclure que la plupart des recherches ont été validées à l'aide d'ensembles de données plus anciens. Ces ensembles de données manquent de nouvelles attaques et contiennent des données d'audit réseau déséquilibrées. Une distribution non uniforme des données peut conduire à une formation biaisée des classificateurs ML et ce problème doit être résolu. Le nouvel ensemble de données peut être utilisé pour détecter de nouvelles attaques. Le classificateur RF montre de meilleurs résultats par rapport aux autres classificateurs. Beaucoup de travail est fait sur la classification binaire et il est encore nécessaire de travailler davantage sur la multi-classification.

3. TAXONOMIE DESLASSIFIERS C

Les classificateurs sont divisés en deux méthodes d'apprentissage, à savoir les apprenants paresseux et les apprenants avides [12-15]. La taxonomie des classificateurs est proposée sur la base d'une analyse théorique et est illustrée à la Fig. 2.

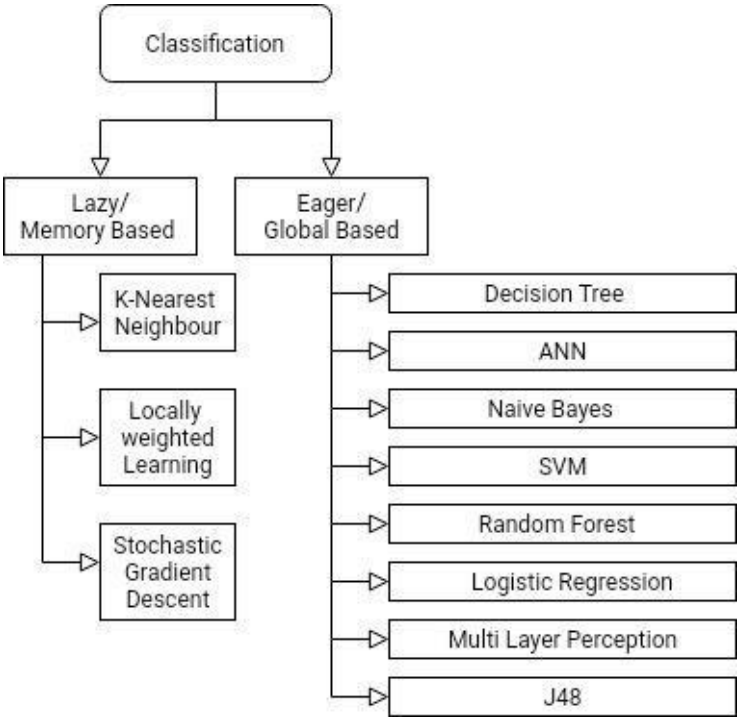

Fig 2: Taxonomie du classificateur

Les apprenants paresseux peuvent stocker des exemples et résoudre plusieurs problèmes avec ces exemples. Ces apprenants s'adaptent automatiquement aux changements dans le domaine du problème et sont faciles à maintenir. Mais la limite de ces apprenants est qu'ils ont stocké le même type d'exemples plusieurs fois et nécessitent donc une mémoire élevée et des apprenants qui prennent beaucoup de temps. Les apprenants enthousiastes construisent d'abord un modèle de classification sur des données de formation données, puis effectuent une classification. Ces apprenants prennent plus de temps pour apprendre et moins de temps pour classer les données.

À partir de la taxonomie des classificateurs ci-dessus, KNN, SGD, DT, NB, RF et LR sont utilisés pour des travaux expérimentaux dans cet article. La description de ces classificateurs qui sont explorés pour des travaux expérimentaux est donnée ci-dessous:

3.1. Apprenants paresseux

Ces apprenants utilisent les données d'apprentissage pour le stockage et attendent que les données de test apparaissent. KNN, Localweighted learning (LWL) et SGD sont des exemples d'apprenants paresseux.

3.1.1. K- Voisin le plus proche

C'est un algorithme d'apprentissage paresseux qui stocke d'abord toutes les données d'entraînement. Au moment de la classification, il utilise ces données et tente de trouver les similitudes entre les nouvelles données et les données disponibles. Il place les nouvelles données dans la catégorie la plus similaire aux données disponibles. Elle est basée sur la distance euclidienne [16]. Les données d'essai sont attribuées à la classe de ses K plus proches voisins. Au fur et à mesure que vous augmentez la valeur de K, la précision peut augmenter. Il peut être utilisé à la fois pour la régression et la classification, mais il est souvent utilisé pour les problèmes de classification.

3.2. Apprenants enthousiastes

Les apprenants enthousiastes prennent beaucoup de temps pour la formation et moins de temps pour prévoir. DT, NB, LR, SVM, RF, MLP et J48 sont des exemples d'apprenants enthousiastes.

3.2.1. Arbre de décision

C'est un outil populaire et puissant pour la prédiction et la classification. La structure de DT est comme une arborescence dans laquelle chaque nœud interne représente un test sur un attribut, chaque branche interprète un résultat du test et chaque nœud feuille affiche une étiquette de classe. DT effectue la classification sans nécessiter beaucoup de calcul et capable de gérer à la fois des caractéristiques catégorielles et continues. Cette arborescence est coûteuse en calcul et montre des erreurs dans les problèmes de multi-classification [17].

3.2.2. Régression logistique

Il est appliqué pour résoudre à la fois les problèmes de classification binaire et multiclasse. La probabilité d'occurrence d'un événement est prédite en donnant des données appropriées à la fonction logistique. La sortie de cette fonction se situe entre 0 et 1. La valeur médiane, c'est-à-dire 0,5, est considérée comme le seuil entre la classe 1 et la classe 0. La puissance supérieure à 0,5 est considérée comme appartenant à la classe 1 et si la production est inférieure à 0,5, elle est considérée comme appartenant à la classe 0[6].

3.2.3. Forêt aléatoire

Il est proposé par Breiman en 2001. Cette méthode est basée sur la recherche de proximité et peut être utilisée à la fois pour la régression et la classification. Il s'agit d'un classificateur basé sur un arbre de décision. Dans cette technique, des échantillons aléatoires sont utilisés pour créer des arbres de décision, puis la prédiction est faite à partir de chaque arbre et la meilleure solution est trouvée par vote [16]. La forêt aléatoire a de nombreuses applications telles que la classification d'images, la sélection de fonctionnalités et les moteurs de recommandation.

3.2.4. Bayes naïfs

C'est un algorithme de classification utilisé à la fois pour les problèmes de classification à deux classes et multi-classes. Il suppose que les probabilités de chaque caractéristique appartenant à chaque classe sont utilisées pour la prédiction [6]. Il suppose également que la probabilité que chaque entité appartienne à une valeur de classe donnée est indépendante des autres entités. Pour la valeur connue de l'entité, la probabilité est appelée probabilités conditionnelles. La prédiction peut être obtenue en calculant les probabilités d'instance de chaque classe et en sélectionnant la valeur de classe la plus probante [12].

4. ENSEMBLE DE DONNÉES USED

Les ensembles de données de référence utilisés dans la littérature sont des ensembles de données plus anciens et contiennent des enregistrements répétés en raison desquels les classificateurs de ML donnent des résultats injustes. Ainsi, certains classificateurs ML sont testés à l'aide de l'ensemble de données UNSW-NB15 qui est un nouvel ensemble de données [18]. Ce jeu de données est composé de 49 attributs, y compris une étiquette de classe, et contient 25 40 044 instances étiquetées, chacune étant étiquetée normale ou attaque. Une description détaillée des caractéristiques est donnée dans le tableau 1.Le tableau 2 donne les détails des attaques.

Tableau 1 : Description des attributs de l'ensemble de données UNSW-NB15

S.No.	Type de Attributs	Nom des attributs	Séquence Non.
1	Couler	Script, Sport, Dstip. Dsport, Proto	1-5
2	Basique	état, dur, sbytes, dbytes, sttl, dttl, sloss, dloss, service, sload, Dload, Spkts, Dpkts	6-18
3	Contenu	Swin, Dwin, Stepb, Dtcpb, Smeansz, Dmeansz, trans_depth, res_bdy_len	19-26
4	Heure	Sjit, Djit, Sestimate, Ltime, Sintpkt, Dintpkt, Tcprtt, Synack, Ackdat	27-35
5	Usage général	is_sm_ips_ports, ct_state_ttl, ct_flw_http_mthd is_ftp_login ct_ftp_cmd	36-40
6	Connexion	ct_srv_src , ct_srv_dst , ct_dst_ltm, ct_src_ ltm ,ct_src_dport_ltm,ct_dst_sport_ltm,ct_dst_src_ltm	41-47
7	Étiquetés	attack_cat, Étiquette	48-49

Lorsque l'ensemble de données UNSW-NB15 est utilisé pour l'évaluation, sur 49 attributs, nous n'avons obtenu que 45 attributs. Les quatre attributs ID sont combinés ensemble pour créer un seul attribut en tant qu'ID de la catégorie d'attribut de flux et deux attributs de la catégorie de temps (Stime et Ltime) sont combinés ensemble dans un attribut appelé Rate. Les 42 attributs de l'ensemble de données UNSW-NB15 sont utilisés pour mener des expériences sur des classificateurs ML. Les attributs d'abandon sont ID, Durée et attack_cat.

Tableau 2 : Types d'attaques dans l'ensemble de données

Type	**Entier**	**Formation**
	Non. Nombre de dossiers	**Non. Nombre de dossiers**
Normal	2218761	56000
Fuzzers	24246	18184
Analyse	2677	2000
Portes dérobées	2329	1746
DOS (en anglais)	16353	12264
Exploits	44525	33393
Générique	215481	40000
Reconnaissance	13987	10491
ShellCode	1511	1133
Vers	174	130
		175341

5. MÉTHODOLOGIE

Les étapes de prétraitement sont illustrées à la Fig.3 et la Fig. 4 montre la méthodologie utilisée. En prétraitement, tout d'abord les valeurs nulles présentes dans le jeu de données sont gérées. Les données catégorielles sont converties sous forme numérique à l'aide d'un codeur d'étiquettes. Ensuite, un codeur à chaud est utilisé pour rompre la relation entre les valeurs obtenues grâce au codeur d'étiquettes.

Après cela, les données prétraitées sont séparées en tant que formation et test. Les classificateurs KNN, LR, NB, SGD, DT et RF sont utilisés pour construire les modèles. Ensuite, la prédiction des étiquettes des données de test est effectuée à l'aide de ces modèles. Une comparaison est effectuée entre les étiquettes réelles et les étiquettes prévues. Les paramètres de performance utilisés pour évaluer les modèles sont l'exactitude, la précision, l'erreur quadratique moyenne, le rappel, le score f1, le TPR et le FPR.

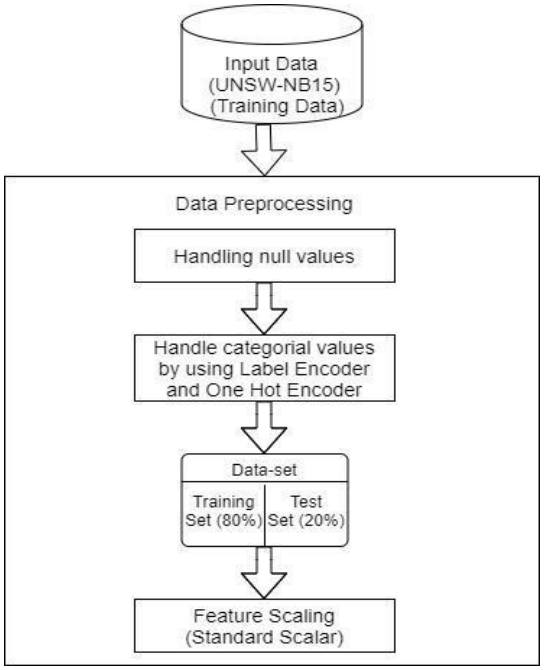

Fig. 3 Étapes de prétraitement sur l'ensemble de données UNSW-NB15

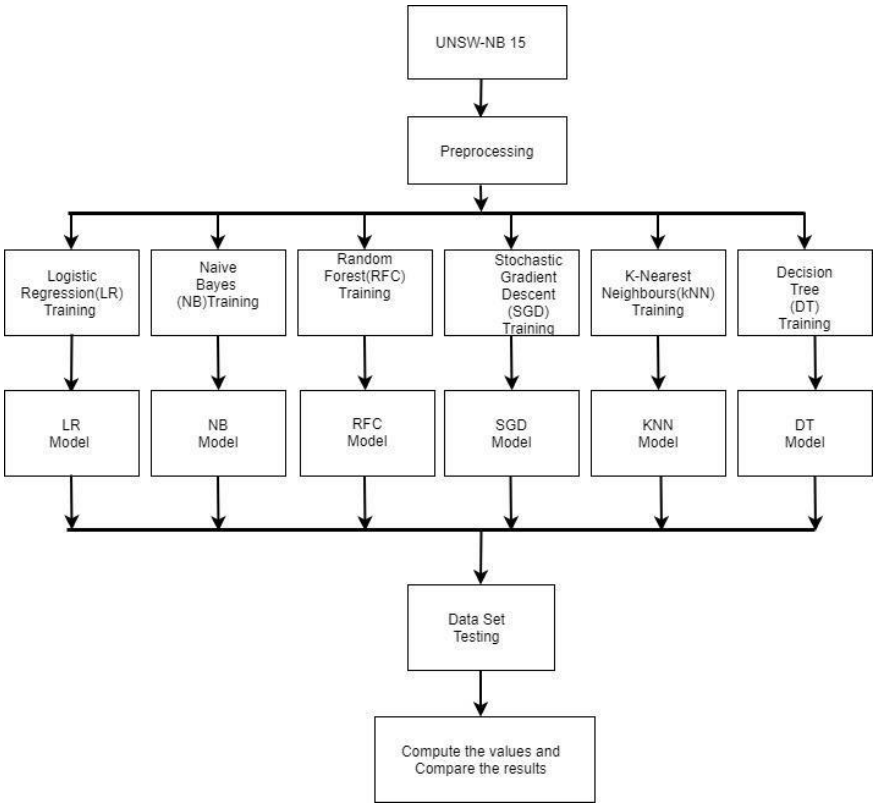

Fig. 4 Méthodologie

Les étapes procédurales pour construire les modèles sont indiquées ci-dessous:

1. Commencez par le prétraitement du jeu de données.
2. Divisez l'ensemble de données en deux parties, c'est-à-dire la formation et les tests.
3. Construire le modèle de classificateur à l'aide de données d'apprentissage pour KNN, LR, NB, RF, SGD et DT.
4. Prenez les données de test
5. Test de modèles de classificateurs à l'aide de données d'apprentissage
6. Calculez et comparez l'exactitude, le rappel, la précision, le score F1 et l'erreur quadratique moyenne pour les modèles sélectionnés.

6. TRAVAIL EXPÉRIMENTAL

Les classificateurs ML sélectionnés, à savoir LR, NB, RF, SGD, KNN et DT, sont testés sur l'ensemble de données UNSW-NB15, le nouvel ensemble de données pour la détection des intrusions. Le travail expérimental est effectué sur Intel Core (TM) i3-1005G1 CPU @ 1.20 GHz, 4GB RAM en utilisant Python. Après avoir effectué les étapes de prétraitement, le jeu de données est divisé en deux parties : la formation et les tests. Ensuite, six classificateurs sont utilisés pour la formation comme le montre la figure 4 et les performances sont évaluées sur la base de plusieurs paramètres, comme le montre le tableau 3. La figure 5 montre la représentation picturale de la précision des classificateurs sélectionnés.

On peut observer, d'après les résultats présentés dans le tableau 3, que le classificateur RF surpasse les autres méthodes en termes de précision 95,43 %, FPR 0,08 et d'erreur quadratique moyenne de 0,046, tandis que le NB affiche l'erreur quadratique moyenne la plus élevée de 0,519 et la précision la plus faible de 48,03 % dans le groupe sélectionné de classificateurs.

Tableau 3 : Comparaison des performances de classificateurs sélectionnés à l'aide de l'ensemble de données UNSW-NB15 avec la méthode de répartition des essais de train

Classificateur	**Exactitude**	**Précision**	**Rappeler**	**F1-Score**	**Erreur quadratique moyenne**	**TPR**	**Le**
LR	93.23	0.92	0.99	0.95	0.068	0.99	0.19
NB	48.03	1.00	0.23	0.38	0.519	0.23	0
RF	95.43	0.96	0.97	0.97	0.046	0.97	0.08
SGD	93.29	0.91	1.00	0.95	0.067	0.99	0.21
KNN	93.71	0.94	0.96	0.95	0.063	0.96	0.12
DT	94.20	0.93	0.98	0.96	0.058	0.98	0.14

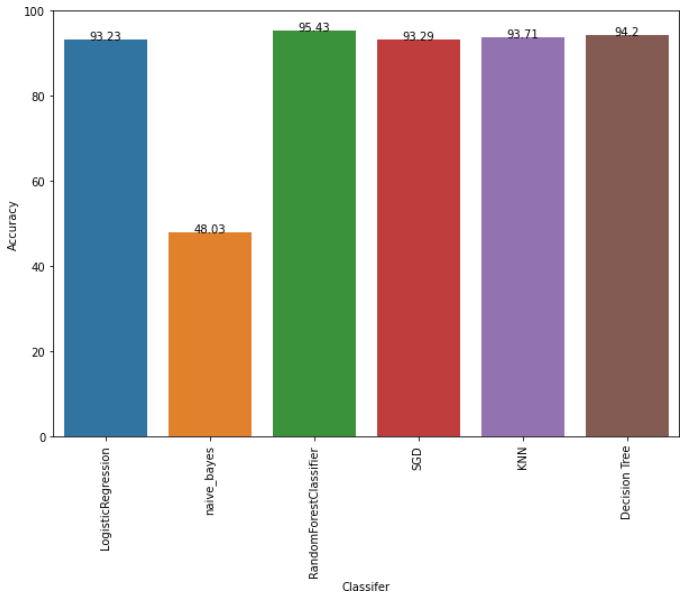

Fig. 5: Précision des classificateurs sélectionnés à l'aide de l'ensemble de données UNSW-NB15 avec test de train Méthode de division

7. CONCLUSION ET AVENIR SFAIRE FACE

La taxonomie des classificateurs est proposée en termes d'apprenants paresseux et avides. Le travail expérimental a été effectué pour évaluer la performance des classificateurs ML sélectionnés sur la base de la taxonomie proposée, à savoir KNN, LR, NB, DT, SGD et RF pour la détection d'intrusion. Ces classificateurs sont testés sur l'ensemble de données UNSW-NB15. Les classificateurs sont comparés sur la base de la précision, MSE, rappel, F1-Score, exactitude, TPR et FPR. Les résultats montrent que le classificateur RF est meilleur que les autres classificateurs de l'ensemble de données UNSW en utilisant des paramètres sélectionnés. La précision du classificateur RF est de 95,43%. À l'avenir, ce travail pourra être étendu aux attributs sélectifs et à la classification multiclasse pour la détection des intrusions.

RÉFÉRENCES

1. Sarmah, A. (2001). Systèmes de détection d'intrusion : définition, besoin et défis.
2. Omer, K. A. A., & Awn, F. A. (2015). Évaluation des performances des systèmes de détection d'intrusion à l'aide d'ANN. *Egyptian Computer Science Journal*, *39*(4).
3. Diro, A. A., et Chilamkurti, N. (2018). Schéma de détection d'attaque distribué utilisant une approche d'apprentissage profond pour l'Internet des objets. *Future Generation Computer Systems*, *82*, 761-768.
4. Khan, J. A. et Jain, N. (2016). Une enquête sur les systèmes de détection d'intrusion et les techniques de classification. *Int. J. Sci. Res. Sci., Eng. Technol.*, *2*(5), 202-208.
5. Narudin, F. A., Feizollah, A., Anuar, N. B., et Gani, A. (2016). Évaluation des classificateurs d'apprentissage automatique pour la détection des logiciels malveillants mobiles. *Soft Computing*, *20*(1), 343-357.
6. Belavagi, M. C. et Muniyal, B. (2016). Évaluation des performances des algorithmes d'apprentissage automatique supervisés pour la détection des intrusions. *Procedia Computer Science*, *89*, 117-123.
7. Ashfaq, R. A. R., Wang, X. Z., Huang, J. Z., Abbas, H., & He, Y. L. (2017). Approche d'apprentissage semi-supervisé basée sur le flou pour le système de détection d'intrusion. *Sciences de l'information*, *378*, 484-497.
8. Al-Yaseen, W. L., Othman, Z. A. et Nazri, M. Z. A. (2017). Machine à vecteur de support hybride à plusieurs niveaux et machine d'apprentissage extrême basée sur des moyens K modifiés pour le système de détection d'intrusion. *Expert Systems with Applications*, *67*, 296-303.
9. Aljumah, A. (2017). Détection des attaques par déni de service distribué à l'aide de réseaux neuronaux artificiels. *International Journal of Advanced Computer Science and Applications (IJACSA)*, *8*(8).

10. Roshan, S., Miche, Y., Akusok, A., & Lendasse, A. (2018). Système de détection d'intrusion réseau adaptatif et en ligne utilisant le clustering et les machines d'apprentissage extrême. *Journal of the Franklin Institute*, *355*(4), 1752-1779.
11. Ali, M. H., Al Mohammed, B. A. D., Ismail, A., et Zolkipli, M. F. (2018). Un nouveau système de détection d'intrusion basé sur le réseau d'apprentissage rapide et l'optimisation des essaims de particules. *IEEE Access*, 6, 20255- 20261.
12. Bhavani, D.D., Vasavi, A. & Keshava P.T. (2016). Machine Apprentissage : examen critique des techniques de classification. IJARCCE, 5(3), 22-28.
13. Wei, C. C. (2015). Comparer des modèles d'apprentissage paresseux et avides pour la prévision des niveaux d'eau dans les bassins fluviaux-réservoirs des régions inondées. Environmental Modelling & Software, 63, 137-155.
14. Rafatirad, S. et Heidari, M. (2018). Une analyse exhaustive des méthodes d'apprentissage paresseuses et avides pour l'investissement immobilier immobilier.
15. https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623
16. Narudin, F. A., Feizollah, A., Anuar, N. B., et Gani, A. (2016). Évaluation des classificateurs d'apprentissage automatique pour la détection des logiciels malveillants mobiles. Soft Computing, 20(1), 343-357.
17. https:// www.geeksforgeeks.org/decision-tree/
18. Moustafa, N. et Slay, J. (2016). L'évaluation des systèmes de détection des anomalies de réseau: analyse statistique de l'ensemble de données UNSW-NB15 et comparaison avec l'ensemble de données KDD99. Information Security Journal: A Global Perspective, 25(1-3), 18-31

AUTEURS

Mme Geeta Kocher

Elle est MCA, M.Tech, M.Phil. Elle poursuit un doctorat dans le domaine de l'intelligence artificielle - apprentissage profond. Elle a publié plus de 15 articles dans diverses conférences et revues. Elle a plus de 16 ans d'expérience dans l'enseignement.

Dr Gulshan Kumar

Dr. Gulshan Kumar a obtenu son diplôme de MCA de l'Université Guru Nanak Dev Amritsar (Pendjab) Inde en 2001, et M.Tech. Diplôme en informatique et ingénierie de JRN Rajasthan Vidyapeeth Deemed University, Udaipur (Rajasthan) - Inde, en 2009. Il a obtenu son doctorat de l'Université technique du Pendjab, Jalandhar (Pendjab) - Inde. Il a 17 ans d'expérience dans l'enseignement. Il a 56 publications internationales et nationales à son actif. Actuellement, il travaille comme professeur agrégé en informatique Département des applications au Shaheed Bhagat Singh State Technical Campus, Ferozepur (Pendjab)-Inde. Il a supervisé 06 étudiants M. Tech. pour leur thèse finale, des étudiants pour des projets MCA et superviseur 02 doctorants chercheurs. Ses intérêts de recherche actuels portent sur l'intelligence artificielle, la sécurité des réseaux, l'apprentissage automatique et les bases de données.