Université paris 8

PARIS 8 UNIVERSITE VINCENNES - SAINT-DENIS

LIASD LABORATOIRE INFORMATIQUE AVANCÉE DE SAINT-DENIS Université Paris 8

FICHE DE LECTURE

pour le cours « Techniques d'apprentissage artificiel »

sur :

Supervised Machine Learning for Intrusion Detection

Une synthèse de plusieurs articles de recherches sur le domaine de détection d'intrusions

(Ahmad et al., 2018) (H. Wang, Gu et S. Wang, 2017) (Liu et al., 2021) ...

faite par

Yasser RABHI

le 11 janvier 2022

Master mention *Informatique* Option *Big data*

Université Paris 8 Vincennes à Saint Denis

COMUE Paris Lumières
Laboratoire Paragraphe
Laboratoire d'Informatique Avancée de Saint Denis

MASTER

Table des matières

1 Introduction

Avec le développement vaste et rapide des nouvelles technologies et leurs applications ; tel que la 5G, l'internet des objets, le Cloud etc, le trafic en temps réel devient de plus en plus compliqué et les cybers attaques augmentent en nombre et en efficacité.(Liu et al., 2021)

Cela entraîne pleins de défis pour les spécialistes du domaine de la cybersécurité. Par conséquent, ces experts ont besoin de développer divers systèmes de détection d'intrusions afin de protéger les systèmes informatiques de toute forme d'attaque.

La détection d'intrusions est une des grandes problématiques du domaine de la cybersécurité telle que les systèmes de détection d'intrusion, les systèmes de prévention d'intrusion et les pare-feux.

Plusieurs techniques sont disponibles pour la détection d'intrusion mais le plus grand souci est en matière de précision et d'efficacité.(Ahmad et al., 2018)

Les techniques d'apprentissage profond s'avèrent très capables de découvrir les modèles d'intrusion compliqués après l'évaluation des caractéristiques représentées par le trafic de réseau.

2 Problématique

2.1 Objectif

Le système de détection d'intrusion dans le réseau doit précisément identifier les attaques malveillantes, formuler des stratégies et fournir un suivi en temps réel et des mesures de protection dynamiques contre ces attaques.(Liu et al., 2021)

L'objectif ici est d'exploiter les caractéristiques des données dimensionnelles à travers les modèles d'apprentissage et convertir le problème de détection d'anomalie dans le réseau vers un problème de classification.

Une analyse est fait sur un ensemble de données afin de comparer les performances de quelques algorithmes de classification.

2.2 Description des données

Dans un cas pratique, les données seront récupérées directement du système d'information à travers les fichiers de journalisation (logs) par exemple. Vu la complexité des systèmes, les données nécessitent un grand travail de test, évaluation et ajustement avant d'être déployés.

En l'occurrence, tout ces articles utilisent NSL-KDD : "NSL–knowledge discovery and data mining (KDD) dataset", la base de données classique dans le domaine de détection d'intrusion (FIG. 1). Cette data-set se composent de vraies traces de réseaux étiquetées avec un vaste ensemble d'intrusions et comportements inhabituels. Ces données sont rigoureusement analysées et nettoyés avant leurs utilisations dans notre algorithme.

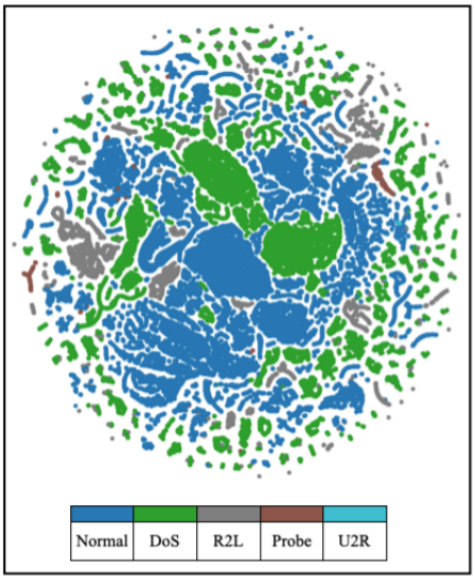

Fig. 1 : t-SNE pour visualiser **NSL-KDD** (Liu et al., 2021)

Dans cette figure (FIG. 1), l'agorithme t-SNE est utilisé afin de visualiser les données de **NSL-KDD** via une réduction de dimentionnalité.

On peut remarquer que les exemples normaux sont plus nombreux que les exemples d'attaques, ce qui rend les attaques plus facile à se cacher et la classification plus indiscernable.

2.3 Difficultés à résoudre

La protection des données sur les ordinateurs ou les réseaux est primordiale pour les individus ainsi que les entreprises , les informations compromises peuvent causer des énormes dégâts aux systèmes informatiques en question.

Les systèmes de détection d'intrusion doivent être le plus efficace possible envers toute sorte d'attaque connue or inconnue.

La précision d'un algorithme de classification dépend du taux détection et du taux de fausses alarmes.

Le problème de précision dans la classification doit être adressé afin de réduire le taux de fausses alarmes (False Alarm Rate) et d'augmenter le taux de détection (Detection Rate).

Cette notion est fondamentale dans nos articles de recherche.

3 État de l'art

Récemment, plusieurs algorithmes d'apprentissage artificiel ont été proposés pour améliorer les performances des systèmes de détection d'intrusion.

- Wang et Al. (H. Wang, Gu et S. Wang, 2017) proposent une infrastructure logicielle de détection d'intrusion basée sur les machines à vecteurs de support (SVM) et validée sur la même base de données **NSL-KDD**. Cette méthode proclame un taux d'efficacité de 99.92% qui est supérieur aux autres méthodes. Mais, les performances des SVMs diminuent pour les bases de données de grande taille. Donc, ce n'est pas le meilleur choix lors de l'analyse d'un immense trafic réseau.

- Raman et al (Gauthama Raman et al., 2017) proposent un mécanisme de détection d'intrusion basé sur un algorithme génétique d'hypergraphes (HG-GA) pour la sélection de caractéristiques dans SVM. Cette méthode indique un taux de détection de 97.14% sur la base **NSL-KDD**.

- Farnaaz et Jabbar (Farnaaz et Jabbar, 2016) développent un modèle de détection d'intrusion basé sur les forêts d'arbres décisionnels (Random Forest). Ils testent l'efficacité de leur méthode sur la même data-set **NSL-KDD**. Malgré un taux de détection de 99.67%, cette proposition n'est pas optimale vu les faibles performances en ordre de temps d'exécution. Les arbres décisionnels ne sont pas adéquates à la résolution d'un problème en temps réel. Cette solution est très lente à cause de la formation d'un grand nombre d'arbres.

4 Algorithmes

Les mots-clés du modèle proposé dans la figure (FIG. 2) sont la base de données, les prétraitements, la classification, et l'évaluation des résultats. Chaque phase du système proposé est très importante avec une grande influence sur les performances de l'algorithme.

Le but ainsi est de comparer les performances de différents algorithmes de classification, notamment : Support Vector Machine **(SVM)**, Random Forest **(RF)** et Extreme Learning Machine **(ELM)**.

4.1 Pré-traitement

Nos algorithmes de classification ne pourraient pas analyser les données dans leur format d'origine. Un prétraitement est donc nécessaire à cause des valeurs symboliques.

Les valeurs non numériques sont soit éliminées soit remplacées parce qu'ils n'indiquent aucune participation dans la détection d'intrusion. Or, ce processus allonge la phase d'apprentissage et l'architecture de notre algorithme devient plus compliquée avec une perte de mémoire et un long temps

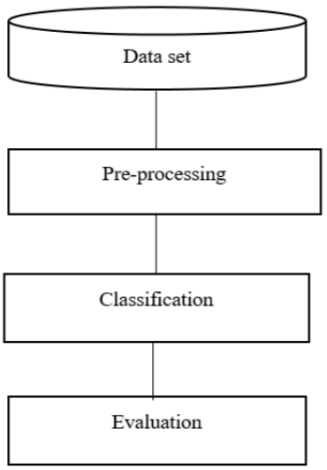

FIG. 2 : Le modèle proposé pour le système de détection d'intrusion (AHMAD et al., 2018)

d'exécution. Afin de pouvoir utiliser l'algorithme en temps réel, ces caractéristiques non numériques sont exclus de la base **NSL-KDD** afin d'améliorer les performances de la détection.

Dans cette figure (FIG. 3) est la liste des caractéristiques de notre base de données **NSL-KDD**.

41 features and one label		
duration	su_attempted	same_srv_rate
protocol_type	num_root	diff_srv_rate
service	num_file_creations	srv_diff_host_rate
flag	num_shells	dst_host_count
src_bytes	num_access_files	dst_host_srv_count
dst_bytes	num_outbound_cmds	dst_host_same_srv_rate
land	is_host_login	dst_host_diff_srv_rate
wrong_fragment	is_guest_login	dst_host_same_src_port_rate
urgent	count	dst_host_srv_diff_host_rate
hot	srv_count	dst_host_serror_rate
num_failed_logins	serror_rate	dst_host_srv_serror_rate
logged_in	srv_serror_rate	dst_host_rerror_rate
num_compromised	rerror_rate	dst_host_srv_serror_rate
root_shell	srv_rerror_rate	label

FIG. 3 : Description des caractéristiques de **NSL-KDD** (PU et al., 2021)

4.2 Classification

Classifier les activités en cours dans un réseau est la principale tache d'un système de détection d'intrusion.

Divers algorithmes de classification multi-classe ont étés utilisés tels que le perceptron multi-couche, les SOMs et les arbres de décision.

Dans cette étude, trois différents algorithmes sont appliqués grâce à leur capacité avérée sur les problèmes de classification.

4.2.1 Support Vector Machine

Proposé initialement par Vapnik en 1995, **SVM** est très utile pour résoudre les problèmes de classification et de régression.

Cette technique d'apprentissage artificiel supervisée présente des avantages uniques en détection de modèles multidimensionnels. L'idée de base est de trouver une séparation entre les différentes catégories.

SVM crée un ou plusieurs hyperplans dans un espace multi-dimensionnel, et le meilleur hyperplan est celui qui divisent les données en plusieurs classes avec un grande séparation entre classes.

SVM a été largement utilisé dans le traitement d'image et la reconnaissance de formes.

La figure (FIG. 4) nous montre l'architecture de ce modèle.

Le noyau RBF utilisé calcule la distance euclidienne entre deux vecteurs numériques et organise les données d'entrée vers un espace multi-diemnsionnel. Le but est de séparer les données d'origines à leurs classes d'attaques respectives.

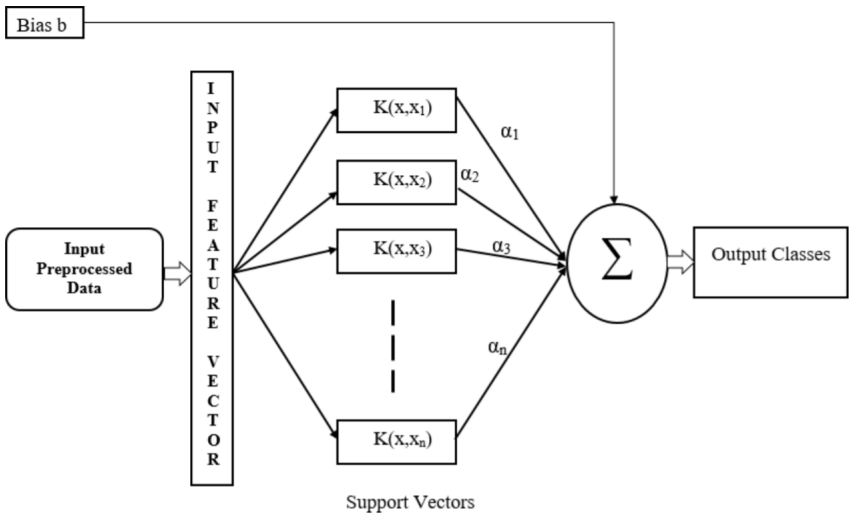

FIG. 4 : Architecture de **SVM** pour la détection d'intrusion (AHMAD et al., 2018)

L'avantage de **SVM** est le petit nombre de paramètres à régler.

4.2.2 Random Forest

Random Forest est un ensemble de classificateurs. **RF** fonctionne en créant plusieurs arbres de décision lors de la phase d'apprentissage . Le label résultant est basé sur la majorité des votes de ces arbres.

Ce modèle peut atteindre des hautes précisions car il peut s'occuper des problèmes comme le bruit et les valeurs aberrantes.

RF est utilisé dans cette étude parce qu'il est moins susceptible d'un surapprentissage et il a récemment montré de bonnes résultats lors de la classification.

Parmis les avantages de Random Forest sont la haute précision et les faibles chances de surapprentissage.

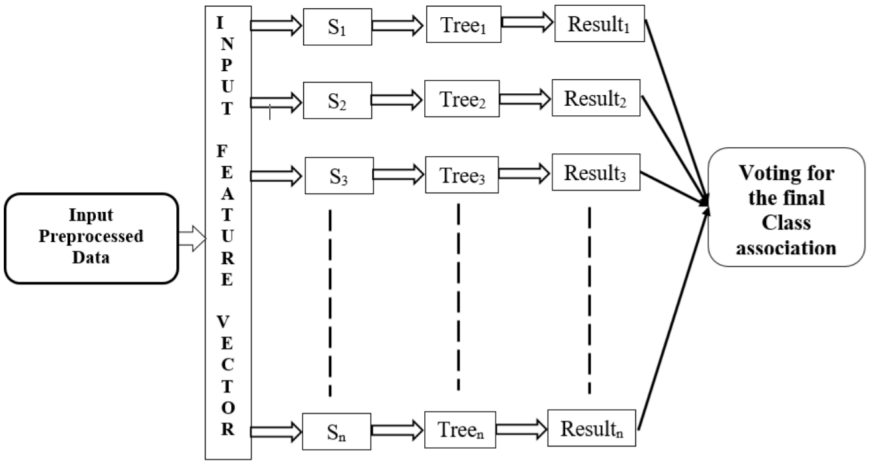

FIG. 5 : Architecture de **RF** pour la détection d'intrusion (AHMAD et al., 2018)

La figure (FIG. 5) nous montre l'implémentation du modèle de classification Random Forest. L'entrée est un ensemble préretraité de **n** exemple. **RF** ensuite crée **n** différentes arbres en utilisant des sous ensemble de caractéristiques. Chaque arbre de décision produit un résultat de classification et le résultat du modelé dépends de la majorité des votes.

4.2.3 Extreme Learning Machine

ELM est une autre appellation pour les réseaux de neurones multi-couches.

Extreme Learning Machine peut être utilisé pour résoudre plusieurs problèmes de classification, clustering, regression et feature engineering.

Cette méthode d'apprentissage combine des couches d'entrées, une ou plusieurs couches intermédiaires cachées et une couche de sortie.

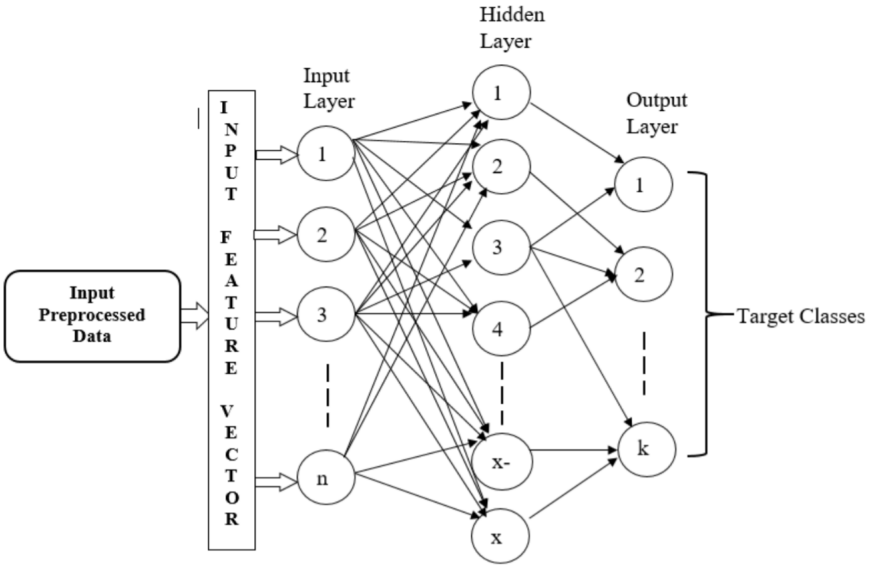

Fig. 6 : Architecture de **ELM** pour la détection d'intrusion (Ahmad et al., 2018)

L'architecture de cette méthode est montrée dans la figure (FIG. 6).

Dans les réseaux de neurones traditionnels, la tâche d'ajustement des poids de la couche d'entrée ainsi que les couches intermédiaires est très couteuse. Afin de surmonter ce problème, l'algorithme choisit arbitrairement les poids d'entrée et ajoute des biais aux couches cachées pour minimiser le temps d'apprentissage.

5 Résultats

Pour mesurer les performances de ces algorithmes, on utilise la matrice de confusion qui compare les résultats actuels avec les résultats prédits.
Cette méthode très connue est utile pour évaluer la précision des prédictions de classification.

5.1 Matrice de confusion

Les algorithmes sont comparés au niveau de leur Accuracy, Precision et Recall.

Soit TP le taux des vraies positives, TN le taux des vraies négatives, P l'ensemble des données positives et N des données négatives :
Accuracy = TP + TN / (P + N)

Precision = TP / (TP + FP)
Recall = TP / P

5.2 Comparaison

Les performances de **ELM** sont comparées avec **SVM** et **RF**. Les résultats de Accuracy, Precision

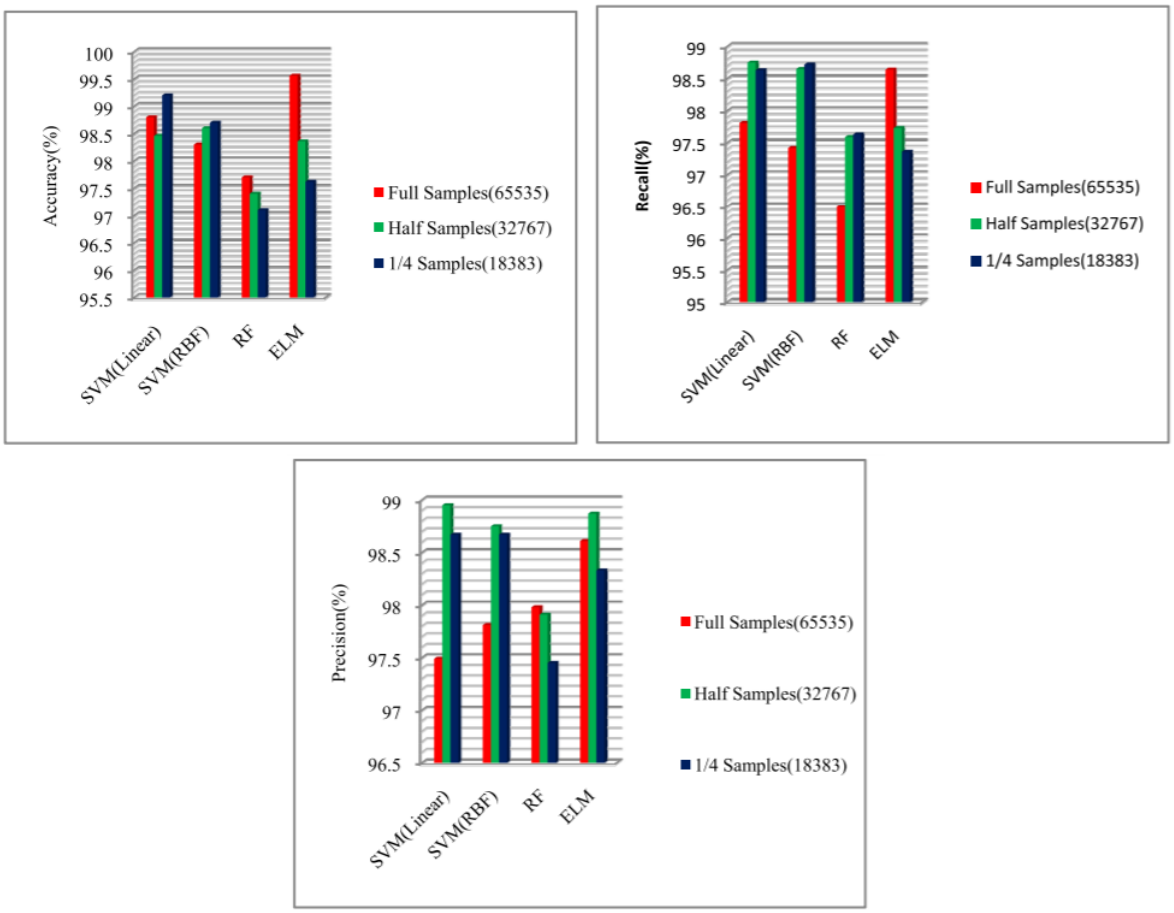

FIG. 7 : Accuracy, Precision et Recall (AHMAD et al., 2018)

et Recall sont montrés dans la figure (FIG. 7). L'ensemble de données est divisé pour avoir 80% de données d'apprentissage et 20% de données de test.

ELM donne de meilleurs résultats par rapport à **SVM** (linéaire), **SVM** (RBF), et **RF** sur la totalité de l'ensemble de données.

Mais, **SVM** donnee des valeurs plus hautes sur le moitié des données et sur un quart des données également. En terme de Recall, **ELM** l'emporte pour un apprentissage sur la totalité des données. Dans les deux autres cas, **SVM** (linéaire) ou **SVM** (RBF) sera plus performant.

La précision de **ELM** est favorable que celles de **SVM** (linéaire et RBF) et **RF** sur la totalité de l'ensemble. Sur le moitié ainsi que le quart des données, les performances de **SVM** en terme de précision sont beaucoup mieux.

6 Conclusion

Les intrusions dans le réseau continuent d'évoluer, la pression sur les systèmes de détection d'intrusion est également de plus en plus.

Les systèmes de détection ou de prévention d'intrusion sont essentiels pour les les réseaux et les courantes systèmes d'informations et celles à venir.
L'apprentissage artificiel propose plusieurs techniques pour améliorer ces systèmes. Ces méthodes varient en matière d'efficacité selon les besoins.

Les résultats observés à travers **SVM** ne sont pas particulièrement satisfaisantes par rapport aux résultats des autres algorithmes que pour de petits ensembles de données.

L'un des désavantages de **SVM** est le besoin d'une fonction pour chaque instance de l'ensemble d'apprentissage. Cela allonge le temps d'apprentissage et dégrade les performances pour les données de plusieurs instances.

Les performances de **ELM** comparées avec **SVM** et d'autres algorithmes d'apprentissage artificiel montrent son pouvoir de fournir de meilleur résultats de classification pour des bases de données de haute complexité.

Dans ce domaine, le choix de caractéristiques (feature engineering) joue un rôle énorme dans la performance de plusieurs algorithmes supervisées.

Bibliographie

[1] Iftikhar Ahmad, Mohammad Basheri, Muhammad Javed Iqbal et Aneel Rahim. *Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection*. 2018. DOI : 10.1109/ACCESS.2018.2841987.

[2] Nabila Farnaaz et M.A. Jabbar. « Random Forest Modeling for Network Intrusion Detection System ». In : *Procedia Computer Science* 89 (2016). Twelfth International Conference on Communication Networks, ICCN 2016, August 19– 21, 2016, Bangalore, India Twelfth International Conference on Data Mining and Warehousing, ICDMW 2016, August 19-21, 2016, Bangalore, India Twelfth International Conference on Image and Signal Processing, ICISP 2016, August 19-21, 2016, Bangalore, India, p. 213-217. ISSN : 1877-0509. DOI : https://doi.org/10.1016/j.procs.2016.06.047. URL : https://www.sciencedirect.com/science/article/pii/S1877050916311127.

[3] M.R. Gauthama Raman, Nivethitha Somu, Kannan Kirthivasan, Ramiro Liscano et V.S. Shankar Sriram. « An efficient intrusion detection system based on hypergraph - Genetic algorithm for parameter optimization and feature selection in support vector machine ». In : *Knowledge-Based Systems* 134 (2017), p. 1-12. ISSN : 0950-7051. DOI : https://doi.org/10.1016/j.knosys.2017.07.005. URL : https://www.sciencedirect.com/science/article/pii/S0950705117303209.

[4] Lan Liu, Pengcheng Wang, Jun Lin et Langzhou Liu. « Intrusion Detection of Imbalanced Network Traffic Based on Machine Learning and Deep Learning ». In : *IEEE Access* 9 (2021), p. 7550-7563. DOI : 10.1109/ACCESS.2020.3048198.

[5] Guo Pu, Lijuan Wang, Jun Shen et Fang Dong. « A hybrid unsupervised clustering-based anomaly detection method ». In : *Tsinghua Science and Technology* 26.2 (2021), p. 146-153. DOI : 10.26599/TST.2019.9010051.

[6] Huiwen Wang, Jie Gu et Shanshan Wang. « An effective intrusion detection framework based on SVM with feature augmentation ». In : *Knowledge-Based Systems* 136 (2017), p. 130-139. ISSN : 0950-7051. DOI : https://doi.org/10.1016/j.knosys.2017.09.014. URL : https://www.sciencedirect.com/science/article/pii/S095070511730415X.

Table des figures