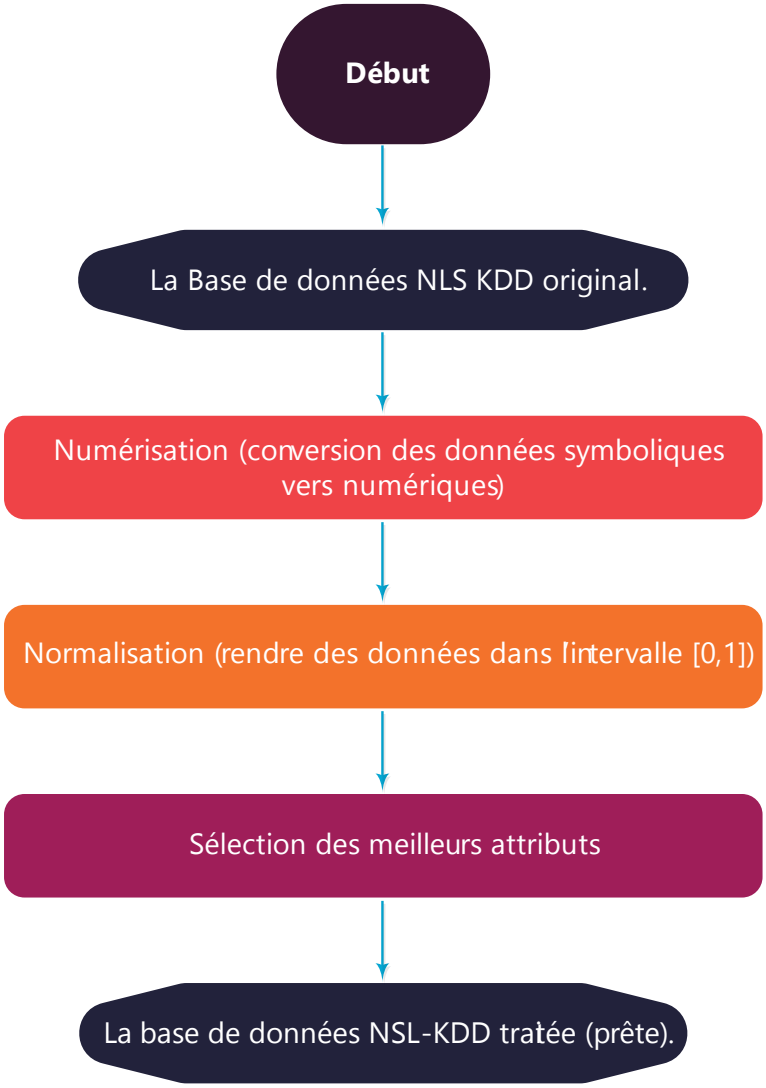
Prétraitement des données



1. La base de données NSL-KDD

Les Données de NSL-KDD sont de trois types: numérique, Nominale et binaire. Les attributs 2, 3 et 4 sont nominales, 7, 12, 14, 15, 21 et 22 sont binaires, et le reste des attributs sont de type numérique, avant de passer au travail expérimental, l'ensemble de données NSL-KDD est d'abord passé par une opération de prétraitement des données et la conversion du type des attributs en suivant les étapes décrites dans ce qui suit.

2. Numérisation

Comme nous l'avons dit précédemment la base de données de NSL-KDD contient 4 colonnes ("protocol_type","Attack","service" et "flag".) ont des données nominales. Sachant que les modèles de prédiction et classification n'acceptent que des attributs numériques. Nous avons converti les quatre colonnes mentionnées ci-dessus vers des données numériques, Il existe plusieurs méthodes de conversion, parmi lesquelles la conversion alphabétique simple que nous avons choisi pour numériser les attributs de type nominal de la base de données NSL-KDD.

Conversion alphabétique simple: La conversion simple consiste à remplacer les valeurs des données catégoriques en ordre alphabétique par des nombres, Les données de l'attribut "type_protocole" par exemple, contiennent trois valeurs catégorielles distinctes : "tcp", "icmp" et "udp". Ces valeurs sont d'abord classées par ordre alphabétique puis on les attribuant un nombre pour chaque catégorie distincte de valeurs.
Et pour faire cette opération on a utilisé la classe **LabelEncoder** de la package **sklearn**.

LabelEncoder : Encodez les attributs cibles avec des valeurs comprises entre 0 et n-1.

Résultat de conversion :

Avant conversion	Après conversion
Icmp	0
Tcp	1
Udp	2

Pour l'attribut "service" qui contient 70 catégories distincts de valeurs, Après l'ordonnancement alphabétique de ces catégories, la valeur de donnée "aol" sera remplacé par "0", "auth" par "1" et "bgp" par "2" et on continue la conversion selon l'ordre alphabétique, comme le montre.

Avant conversion	Après Conversion	Avant conversion	Après Conversion	Avant conversion	Après Conversion
aol	0	http_443	23	printer	46
auth	1	http_8001	24	private	47
bgp	2	imap4	25	red_i	48
courier	3	IRC	26	remote_job	49
csnet_ns	4	iso_tsap	27	rje	50
ctf	5	klogin	28	shell	51
daytime	6	kshell	29	smtp	52
discard	7	ldap	30	sql_net	53
domain	8	link	31	ssh	54
domain_u	9	login	32	sunrpc	55
echo	10	mtp	33	supdup	56
eco_i	11	name	34	Systat	57
ecr_i	12	netbios_dgm	35	Telnet	58
efs	13	netbios_ns	36	Tftp_u	59
exec	14	netbios_ssn	37	Tim_i	60
finger	15	netstat	38	Time	61
ftp	16	nnsp	39	urh_i	62
ftp_data	17	nntp	40	urp_i	63
gopher	18	ntp_u	41	uucp	64
harvest	19	other	42	uucp_path	65
hostnames	20	pm_dump	43	vmnet	66
http	21	pop_2	44	whois	67
http_2784	22	pop_3	45	X11	68
Z39_50	69				

Le même principe est appliqué pour convertir les données de l'attribut Flag

Avant conversion	Après Conversion	Avant conversion	Après Conversion
OTH	0	S1	6
REJ	1	S2	7
RSTO	2	S3	8
RSTOS0	3	SF	9
RSTR	4	SH	10
S0	5		

3. Normalisation:

Les valeurs obtenues après l'opération de la numérisation sont très variées et constituent un grand intervalle, Certains attributs prennent de grandes valeurs (src_bytes, dst_bytes, etc.), alors que d'autres ne prennent que des petites valeurs (serror_rate, same_srvrate, etc.), et cela peut nuire à la rentabilité du modèle de détection d'intrusions.

Afin d'éviter ce problème et garantir l'efficacité du modèle généré, les valeurs de la base données doivent être ajustées ou normalisées ; dans notre cas les données de la base NSL-KDD sont normalisés dans l'intervalle de [0, 1] en se basant sur une fonction de transfert et pour ça on a utilisé la classe **OneHotEncoder** de la package **sklearn**.

OneHotEncoder : Encodez les caractéristiques catégorielles sous la forme d'un tableau numérique unique.
L'entrée de ce transformateur doit être un tableau d'entiers ou de chaînes, indiquant les valeurs prises par les entités catégorielles (discrètes). Les fonctionnalités sont encodées à l'aide d'un schéma de codage one-hot (alias «one-of-K» ou «dummy»). Cela crée une colonne binaire pour chaque catégorie et renvoie une matrice creuse ou un tableau dense (selon le paramètre creuse)
Par défaut, l'encodeur dérive les catégories en fonction des valeurs uniques de chaque entité. Vous pouvez également spécifier les catégories manuellement.

4. La sélection d'attributs :

Vu que la taille de la base de données NSL-KDD est très importante, dans notre cas (41 attributs et 25192 enregistrements pour l'entrainement et 22544 pour le test),donc le travail sur tous les attributs pour générer le modèle de classification sera très fastidieux et peut affecter considérablement les performances de l'algorithme d'apprentissage en matières de temps d'exécution et consommation des ressources systèmes, en plus les attributs de la base de données ne seront pas tous utilisés dans la

classification des connexion TCP/IP effectuée par l'IDS pour détecter les attaques, certains étant plus pertinents que d'autres.

Pour ces raisons, et dans le cadre de notre projet, la sélection des attributs constitue une tache non obligatoire mais très importante pour extraire des sous-ensembles des attributs en préservant les plus significatives et pertinents et en excluant les attributs considérés comme source de bruits dont le but est de faire une classification supervisée des enregistrements de l'ensemble de données NSL-KDD en deux catégories (Normale et Attaque) et de réduire le coût et le temps nécessaire pour l'opération d'apprentissage.