الجمـهورية الجزائرية الديمقراطية الشعبيـة
وزارة التعليم العالي والبحث العلمي
جـامعـة سعيدة. مـولاي الطـاهـر
كـلـيـة التكنولوجيا
قسم: الإعلام الآلي

Mémoire de Master

Spécialité : **Sécurité informatique et cryptographie**

Thème

Détection de maliciels Android : Problème de Sélection d'attributs

Présenté par :

Abdelaziz KEDDARI

Fatma HOUACINE

Dirigé par :

Dr.Mebarka YAHLALI

Année universitaire 2022-2023

Dédicace

Je dédie ce modeste travail : grande mére

À ma source de joie, ceux qui ont toujours veillé sur mon bonheur,

Qui ont sacrifié pour me voir réussi et qui m'ont comblé tant d'amour et de Tendresse

A mes chers parents, mon père (Muhammad), ma mère (Aïcha), que Dieu les protège,

Ils ont été toujours qui m'a soutenu, aimé et guidé dans chaque pas que j'ai fait.

Merci à eux j'ai appris la volonté, le dévouement et la détermination pour réaliser mes rêves. Leurs sacrifices pour que je réussisse dans mes études

Je leur exprime mes remerciements

Pour leur patience sans fin et leurs encouragements constants

Á mes chères sœurs et mes chers frères

A toute ma famille, et à mes chers amis,

Houacine FATMA

Dédicace

Je dédie ce travail à : grande mére

Chers parents, je ne peux qu'exprimer mes sincères sentiments

je les remercie pour leur patience sans fin, leurs encouragements constants et leur aide.

A Mes chers frères et sœurs Qui sont toujours à mes côtés, prêts à m'aider.

A mes chères amies qui sans leurs encouragements ce travail n'aura jamais vu le jour.

A toute ma famille et à tous ceux que j'aime.

je tiens à remercier toutes les personnes qui ont contribué de près ou de loin

De faire ce travail

Keddari abdelAziz

Remerciements

Au nom d'YALLAH le tout Miséricordieux et que la prière et la paix soient sur notre prophète Mohamed aalayh Elssalet Wa Elssalem. Avant tout, le grand et le vrai merci revient à Allah qui nous nous a donné la force, la foi et la vie pour accomplir cette tâche, ainsi que l'audace pour dépasser toutes les difficultés.

Au terme de ces travaux, nous tiens lieu remercier la personne qui nous et aidez nous durant mémoire Notre superviseur Madame ***Mebarka Yahlali*** *Pour sa sagesse, son expérience, ses conseils et encouragements, sa patience, de nous faire confiance, Merci beaucoup.*

Nous remercions chaleureusement les membres de jury pour l'honneur qu'ils nous ont fait en acceptant d'évaluer et d'examiner ce travail.

Je tiens à remercier tous les membres du département d'informatique

Bien entendu, je remercie mes parents et toute la famille pour le soutien moral qu'ils m'ont apporté tout au long de ce travail.

Enfin à exprimer mes sincères remerciements à toutes les personnes qui ont participé de près ou de loin l'exécution de ce modeste travail.

Résumer

La prolifération continue des logiciels malveillants Android suscite de graves préoccupations en matière de sécurité mobile. Incontestablement, la détection des logiciels malveillants Android a reçu beaucoup d'attention dans la communauté de la recherche et devient donc un aspect crucial pour assurer la sécurité des utilisateurs. Cependant, un problème majeur dans ce domaine est la sélection des attributs pertinents pour la détection. Les attributs sont des caractéristiques d'une application Android qui indiquent si elle est malveillante et qui jouent un rôle important dans la fouille de données. La sélection d'attributs permet de représenter un sous ensemble de données à partir d'un ensemble volumineux de données et d'éliminer les données redondantes, non pertinentes. Cependant, il est difficile de déterminer quels attributs sont pertinents en raison de l'évolution rapide des maliciels. De plus, la collecte d'attributs peut être coûteuse en termes de ressources et certains attributs peuvent être trompeurs. Les chercheurs utilisent des techniques de machine Learning et exploitent des bases de données pour extraire automatiquement des attributs pertinents. Malgré les difficultés, il est possible de développer des modèles de détection efficaces en combinant ces approches.

Mots clés : sélection des attributs, la fouille de données, machine Learning, Android

Abstract

The continued proliferation of Android malware raises serious mobile security concerns. Undoubtedly, Android malware detection has received a lot of attention in the research community and hence becomes a crucial aspect to keep users safe. However, a major problem in this area is the selection of relevant attributes for detection. Attributes are characteristics of an Android application that indicate whether it is malicious. They play an important role in data mining. Attributes selection allows to represent a subset of data from a large dataset and to eliminate redundant, irrelevant data.. However, it is difficult to determine which attributes are relevant due to the rapid evolution of malware. Also, collecting attributes can be resource intensive and some attributes can be misleading. Researchers use machine learning techniques and leverage databases to automatically extract relevant attributes. Despite the difficulties, it is possible to develop efficient detection models by combining these approaches.

Keywords: attribute selection, data mining, machine learning. Android

ملخص

يثير الانتشار المستمر لبرامج اندرويد الضارة مخاوف خطيرة تتعلق بأمان الأجهزة المحمولة. مما لا شك فيه إن اكتشاف البرامج الضارة لنظام اندرويد قد حظي باهتمام كبير في مجتمع البحث وبالتالي أصبح جانبا مهما للحفاظ على أمان المستخدمين ومع ذلك فان المشكلة الرئيسية في هذا المجال هي اختيار السمات ذات الصلة للكشف.السمات هي خصائص تطبيق اندرويد تشير إلى ما إذا كانت ضارة أم لا .فهي تلعب دورا مهمًا في التنقيب عن البيانات. يجعل من الممكن تمثيل مجموعة فرعية من البيانات من مجموعة كبيرة من البيانات والتخلص من البيانات الزائدة عن الحاجة وغير ذات الصلة. ومع ذلك ، من الصعب تحديد السمات ذات الصلة بسبب التطور السريع للبرامج الضارة. أيضًا ، يمكن أن يكون جمع السمات مكثفًا للموارد وقد تكون بعض السمات مضللة. يستخدم الباحثون تقنيات التعلم الآلي والاستفادة من قواعد البيانات لاستخراج السمات ذات الصلة تلقائيًا. على الرغم من الصعوبات، من الممكن تطوير نماذج كشف فعالة من خلال الجمع بين هذه الأساليب.

الكلمات الرئيسية: اختيار السمات ، التنقيب عن البيانات ، التعلم الآلي ، اندرو يد

Table des matières

Table des figures

Liste des tableaux

liste des abréviations

ACP analyse en composantes principales

BB Branch and Bound

K-ppv K-plus proches voisins

SFS Séquentiel Forward Sélection

SBS Séquentiel Backward Sélection

LVF Las Vegas Filter

LVW Las Vegas Wrapper

VPN Virtual Private Network

HTTPS Hypertex**t** Transfer Protocol Secure

MITM Man-in –the-Middle

PE portable Executable

KNN K nearest neighbor

API Application programming interface

AA Apprentissage Automatique

GID Group Identifier

UID User Identifier

Wi-Fi Wireless Fidelity

SVM Support Vector Machine

SHA1 Secure Hash Algorithm 1

SSL Secure Sockets Layer

MD5 Message-Digest

Introduction générale

La transmission d'informations et le souci d'assurer la confidentialité de celles-ci est devenue un point primordial et une problématique essentielle que ce soit pour les entreprises, ou pour les individus.
Actuellement, les applications mobiles sont liées aux sites Web, aux systèmes d'information et aux services de stockage de données en nuage, et sont devenues des outils importants dans divers domaines. Android est le système d'exploitation mobile le plus populaire installé sur des millions d'appareils (smartphones, tablettes, téléviseurs, smartwatches). Cette technologie permet l'échange et le partage de ressources et d'informations sensibles via des messages électroniques. Cet effet se traduit par des tentatives de violation des politiques de sécurité par un accès malveillant. Par conséquent, il est important de définir des politiques de sécurité et d'assurer la conformité.
La détection des logiciels malveillants Android est une tâche complexe. Les chercheurs et les développeurs continuent d'affiner les techniques de détection des logiciels malveillants en exploitant divers attributs tels que les autorisations, les comportements anormaux, les signatures et les modèles d'utilisation. Cependant, il est nécessaire de trouver un équilibre entre la précision de détection afin d'éviter les faux positifs et d'améliorer les performances sur les appareils mobiles. Les techniques de réduction dimensionnelle ne sont pas utilisées dans une variété de domaines et d'applications, notamment la science des données, l'apprentissage automatique, l'extraction d'informations, la vision par ordinateur, la visualisation graphique, etc.

L'objectif de travail :
Dans ce contexte notre objectif est d'étudier l'effet de la réduction dimension par sélection d'attributs sur les classifieurs dans le cas des Data set de détection de maliciel Android.
Ce mémoire organisé comme suit :

- **Chapitre 1** :Sélection d'attributs

Ce chapitre présent la sélection d'attributs, ses objectifs, ainsi que les différentes méthodes proposées dans la littérature.

- **Chapitre 2** : Sécurité des applications mobiles

Ce chapitre présente la sécurité des applications mobiles et le modèle de sécurité Android et ses caractéristiques. Ainsi que les autorisations des applications Android, et les travaux et les techniques de détection de maliciel sur Android

- **Chapitre 3 :** Contribution et Implémentation

Nous montrons dans ce chapitre une description générale sur les dataset utilisés, les différents résultat

Chapitre 1 :

Sélection d'attributs

Introduction

La réduction de la dimensionnalité est une discipline de recherche qui vise à réduire le nombre de métriques dans les systèmes de surveillance afin de surmonter les problèmes liés à la collecte et au traitement des données de grande dimension. La réduction de la dimension via l'extraction et la sélection d'attributs est une étape importante dans le traitement des données dans les systèmes de classification. Cette étape peut influer considérablement sur la performance d'un tel système.

La sélection de caractéristiques est un domaine de recherche actif depuis des décennies [1] et a fait ses preuves en théorie et en pratique. Dans le contexte de la classification, l'objectif principal de la sélection d'attributs est de déterminer le sous-ensemble d'attributs de taille minimale qui ne réduit pas de manière significative la précision de la classification.

I. Réduction de dimension

Aujourd'hui, le nombre de métriques collectées par les systèmes surveillés augmente rapidement. Pour cette raison, des recherches intensives ont été menées sur la réduction de la dimensionnalité pour éviter la « malédiction de la dimensionnalité[1] » et réduire le stockage et l'effort de calcul associés au traitement de ces données. La réduction de la dimensionnalité est donc un domaine de recherche L'intersection de plusieurs disciplines, notamment la bioinformatique, la reconnaissance de formes, l'apprentissage automatique, l'intelligence artificielle et l'optimisation. Il existe deux approches générales de la réduction de la dimensionnalité [2,3] :

1. L'extraction d'attributs (Feature Extraction)

Ces techniques permettent de créer de nouveaux ensembles d'attributs, en utilisant des combinaisons (transformation) d'attributs de l'espace initial et en effectuant des modifications générales qui réduisent la dimension.

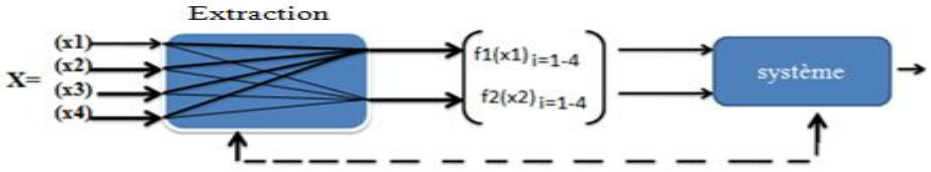


Figure1.1 : La technique d'extraction de caractéristiques [2].

[1]**Malédiction de la dimensionnalité** : est un terme inventé par Richard Bellman en 1961 pour désigner divers phénomènes qui ont lieu lorsque l'on cherche à analyser ou organiser des données dans des espaces de grande dimension

2. La sélection d'attributs (Feature Sélection)

Ces techniques consiste à :

- choisir un sous-ensemble d'attributs de l'espace de variables sans transformation,
- identifier et éliminer les caractéristiques redondantes.

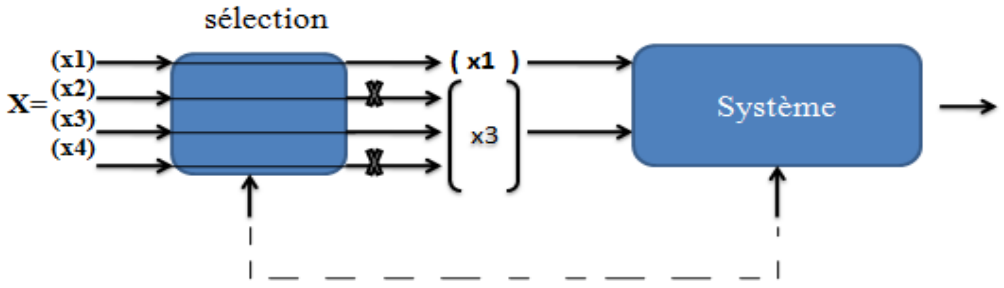


Figure 1.2 : La technique de sélection d'attributs [3].

II. La sélection d'attributs

La sélection d'attributs est un problème difficile qui a été étudie depuis les années 70.Le problème de sélection d'attributs peut être défini comme suit : « Une Technique de prétraitement des données qui consiste à identifier et à supprimer les caractéristiques sans importance ou redondantes dans un ensemble de données. L'objectif est de réduire le nombre de variables tout en préservant les informations importantes. Cette technique est particulièrement utile pour améliorer les performances [4], des modèles d'apprentissage automatique en atténuant les effets de sur ajustement, du bruit et de la complexité du modèle. La sélection d'attributs peut être effectuée de différentes manières, telles que l'analyse de corrélation, l'analyse en composantes principales (ACP), les tests statistiques et les algorithmes d'apprentissage automatique. En fin de compte, sélection d'attributs réduit les coûts de calcul, accélère le traitement et facilite l'interprétation des résultats. »

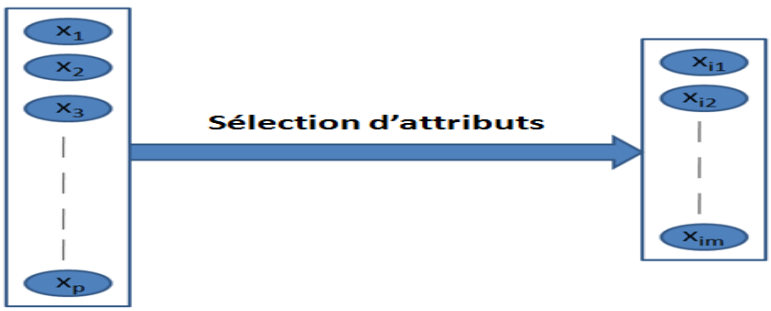


Figure 1.3 : Définition de sélection d'attributs [5].

II.1. Les avantages la sélection des attributs

- Elle réduit le nombre des attributs.
- La sélection des attributs peut augmenter la précision et améliorer les performances du Classificateur.
- La sélection d'attributs (réduit le temps de calcul).

II.2. Objectifs de la sélection d'attributs : Le but de la sélection est donc de trouver les sous-vêtements idéaux à partir d'attributs ayant les propriétés suivantes : Réduction des coûts de collecte d'attributs, amélioration de la précision en mettant en évidence les facteurs associés à la classification, interprétation plus facile des modèles de faible complexité, interprétation plus facile **[6]**.

II.3. Schéma général de la sélection d'attributs :

Les différentes méthodes proposées dans la littérature pour la sélection d'attributs peuvent être décrites par un schéma général dans lequel on trouve les éléments clés suivants (**Figure1.4**):

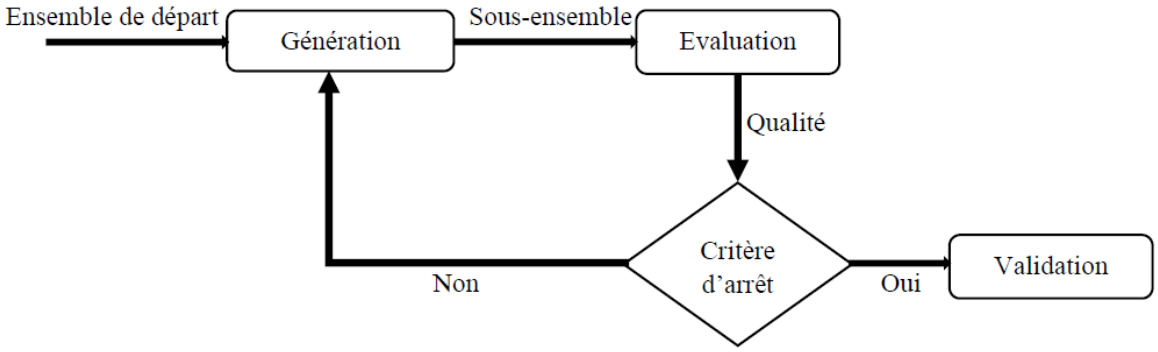


Figure1.4: Schéma général de sélection d'attributs [7].

1. **Génération de sous ensemble :** C'est un processus essentiel de recherche heuristique, avec chaque état de l'espace de recherche fournit une collection de candidats à évaluer. La nature de ce processus est déterminée par les deux étapes : Point de départ, Stratégie de recherche [7].
2. **Evaluation de sous ensemble :** Permet d'évaluer le sous-ensemble généré selon des critères d'évaluation bien précis. Les mesures d'évaluation sont divisées en deux principales catégories (Les mesures indépendantes et Les mesures dépendantes).
3. **Critères d'arrêt :** est utilisé pour décider quand arrêter. Il existe plusieurs critères d'arrêt, on cite : Le nombre des caractéristiques à sélectionner ; Le nombre des itérations.
4. **Validation :** le sous-ensemble choisi doit généralement être validé standard différents tests avec des données du monde réel ou non réel [8].

II.4. Catégorisation des attributs des méthodes de sélection :

Il existe principalement trois grandes approches de sélection :

II.4.1. L'approche par filtre :

Lorsqu'il s'agit de sélectionner les caractéristiques les plus pertinentes pour un algorithme d'apprentissage automatique, le modèle de sélection par filtrage est souvent utilisé. Cette méthode est indépendante de tout algorithme d'apprentissage et consiste à attribuer un score à chaque entité, puis à supprimer celles ayant un score inférieur à un certain seuil. Le sous-groupe d'entités ainsi obtenu est ensuite utilisé pour l'algorithme de classification. Les méthodes de sélection de caractéristiques par filtrage utilisent généralement une approche heuristique pour déterminer les caractéristiques à conserver. La procédure de filtrage est illustrée [9].

-avantage : exécution rapide ; coût de calcul faible.

- inconvénients : Aucune interaction avec le classificateur.

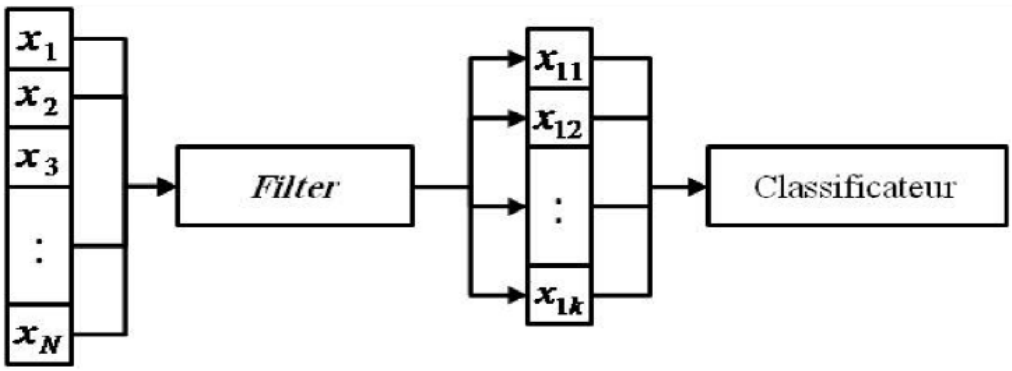


Figure1.5 : La procédure du modèle « filtre » [9].

II.4.2. L'approche wrapper : Le principal inconvénient de la technique du « filtre » provient du fait qu'elle ignore l'effet des attributs sélectionnés sur les performances du classificateur. Cette technique est plus précise car les attributs sélectionnés correspondent bien à l'algorithme d'apprentissage. Cependant, cette méthode présente l'inconvénient d'être plus gourmande en calculs que la méthode de filtrage car elle appelle l'algorithme de classification pour chaque sous-ensemble considéré. [10]

-avantages : Interaction avec le classificateur ; Bonne performance de classification.

- Inconvénients : coût de calcul élevé ; exécution lente.

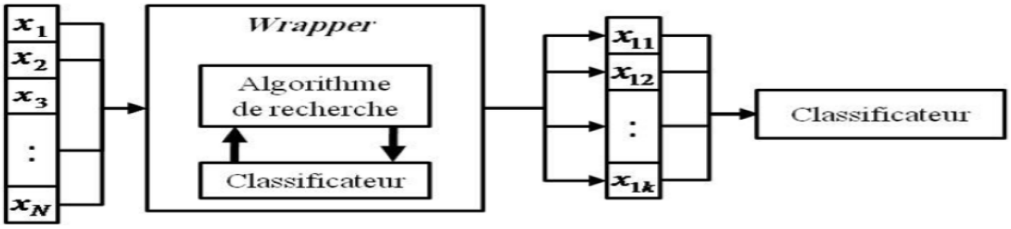


Figure 1.6 : la procédure du modèle « wrapper » [11].

II.4.3. Embedded méthode: Cette technique effectue une sélection en même temps que le processus de classification [12]. Le sous-ensemble optimal d'attributs est déterminé pendant la formation comme suit : Classificateur [13]. Tout comme la technique "wrapper" est la technique "Embedded" Il est spécifique à un algorithme d'apprentissage particulier. Le principal avantage de ceci est La méthode est plus rapide que la méthode "wrapper" [14]. Procédure Le modèle "intégré" est **figure1.7**

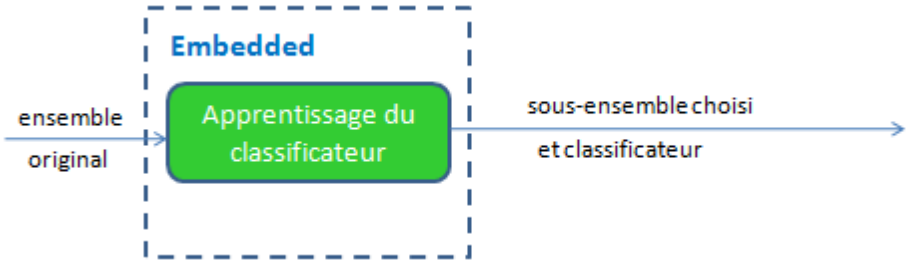


Figure 1.7 : Sélection d'attributs à base Embedded [12]

II.5. Méthodes classiques de la sélection d'attributs :

Pour de nombreuses façons classiques de sélectionner des attributs, Littérature, j'ai décidé d'en présenter quelques-unes, dont la plus célèbre Catégoriser selon la technique utilisée pour la procédure de sous-génération mettre.

II.5.1. Méthodes complètes :

Ces méthodes examinent toutes les combinaisons d'attributs possibles. tu as Très grande complexité (espace de recherche d'ordre O (2N) pour N attributs), mais il garantit de trouver le meilleur sous-ensemble d'attributs. Comme Voici quelques exemples de ces méthodes :

1. FOCUS : Examinez tous les sous-ensembles de variables pour déterminer le plus petit sous-ensemble suffisant pour déterminer l'appartenance à la classe de toutes les instances. Les algorithmes de base définis à l'origine pour les données booléennes sans bruit sont limités à deux classes. Il a une complexité temporelle O(NM). [15]

2. Branch and Bound (BB) : Parcourant l'arbre de la racine à la feuille, l'algorithme supprime successivement les pires propriétés du sous-ensemble courant (nœud courant) qui ne satisfont pas les critères de sélection. Dès que la valeur affectée à un nœud devient inférieure à un seuil (frontière), le sous-arbre de ce nœud est supprimé. L'avantage de cette méthode est qu'elle garantit de trouver le meilleur sous-ensemble de caractéristiques lorsqu'une fonction de notation monotone est utilisée [16].

-Avantages: La valeur métrique calculée pour l'ensemble actuel (a1, a3, a4, a5) peut être inférieure à la limite actuelle. Dans ce cas, aucun sous-ensemble de nœuds * ne peut produire de meilleures valeurs du critère J que la borne B en raison de la condition de monotonie [16].

-Inconvénients : Le calcul de la valeur de critère est généralement plus lent (sous-ensembles d'attributs évalués sont plus grands) ; Les sous-arbres à couper sont moins fréquents près de la racine (valeurs critères plus élevées peuvent être attendues pour les grands sous-ensembles, ce qui réduit le risque pour la valeur du critère d'être inférieur à la meilleure valeur courante).

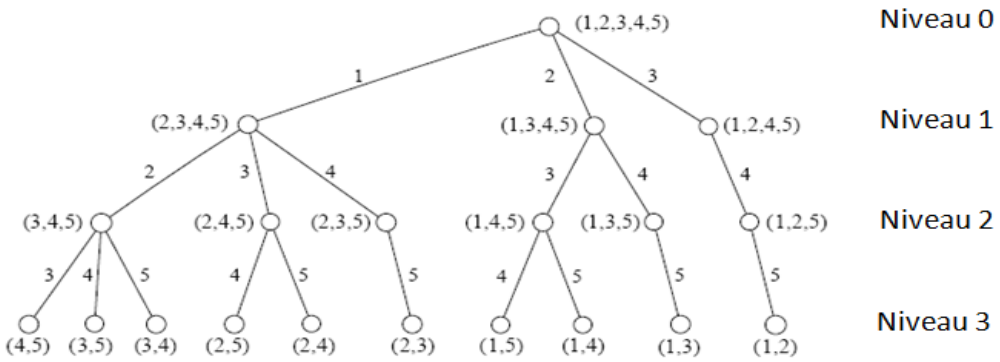


Figure 1.8 :L'arbre solution de l'algorithme BB lorsque m=2 et p=5 [16].

II.5.2. Méthodes heuristiques :

La recherche est effectuée de telle manière qu'elle n'a pas à évaluer tous les ensembles. Un sous-ensemble d'attributs possibles. En conséquence, une accélération est obtenue. Parce que l'espace de recherche est plus petit que la méthode complète. Les méthodes heuristiques ne sont pas

garanties pour trouver le meilleur sous-ensemble. A titre d'exemple, nous introduisons la méthode de sélection directe séquentielle. (SFS), (SBS).

1. SFS : Séquentiel Forward Sélection : Cette méthode est une technique de recherche heuristique apparue en 1963. Commencez avec un ensemble vide et ajoutez des fonctionnalités jusqu'à ce que les critères de terminaison soient remplis. Cette méthode a été utilisée dans [17] pour réduire la taille des données et améliorer les résultats de la classification [18].

2. SBS : Séquentiel Backward Sélection : Cette méthode date de 1971 [19]. Le principe général de cette méthode est de commencer avec l'ensemble complet de toutes les fonctionnalités et de supprimer les fonctionnalités de manière séquentielle. Cette méthode est plus efficace que la méthode précédente (SFS), mais le principal problème est le temps de calcul.

II.5.3. Méthodes Aléatoires :

Ces méthodes n'ont pas de méthode spécifique pour générer des sous-ensembles Analysez les attributs, mais utilisez une méthode aléatoire. Alors cherche La théorie des probabilités s'exécute dans l'espace des caractéristiques. Résultats obtenus avec L'utilisation de ces types de méthodes dépend du nombre d'itérations, qui n'est pas garanti. Atteindre la taille de portion optimale. Méthode LVF et algorithme génétique appartiennent à cette catégorie.

1. las Vegas Filtre et Las Vegas Wrapper : Deux méthodes complètes ont été proposées respectivement en 1996 et 1998 (LVW [20] et LVF [21]). Ces deux méthodes génèrent aléatoirement un sous-ensemble de fonctionnalités pour chaque itération. La méthode LVW utilise un classificateur (méthode d'enveloppe) pour noter des sous-ensembles, contrairement à LVF (méthode de filtrage), qui effectue une notation en calculant une mesure appelée «taux de dissemblance».

2. Les algorithmes génétiques pour la sélection d'attributs : Plusieurs chercheurs proposent l`hybridation de l`algorithme génétique avec d`autre méthodes. IL y a eu une proposition d`une nouvelle méthode hybride pour la sélection des caractéristiques, elle utilise les deux algorithmes Branch and Bound et les algorithmes génétiques pour résoudre le problème de la sélection [22].Une autre approche de sélection des caractéristiques à base d`un algorithme génétique et avec l`utilisation de K-ppv comme étant une fonction fitness [23].

II.6. Applications de la sélection d'attributs :

La sélection d'attributs a trouvé une large applicabilité, car de nombreux systèmes traitent de grandes quantités de données dans divers domaines. Les principaux domaines d'application sont la fouille de données en général, la classification de textes en particulier, la reconnaissance de formes telles que des caractères et des images, et le domaine de la bioinformatique [24].

1. La fouille de données : La fouille de données (data mining) dans de très grandes bases devient un problème critique pour des applications telles que le génie génétique, la finance, les

études de marché, les processus industriels complexes La fouille de données est un domaine basé sur les statistiques, l`apprentissage automatique et la théorie des bases de données. La sélection d`attributs joue un rôle important dans le data mining, en particulier dans la préparation des données pour le traitement [25].

2. La catégorisation de textes : Un problème central pour la catégorisation de textes est la grande dimension de l'espace de représentation. Par exemple, avec la représentation en sac de mots, chacun des mots d'un corpus est un descripteur potentiel ; où pour un corpus de taille raisonnable, ce nombre peut être de plusieurs dizaines de milliers. Pour beaucoup d'algorithmes d'apprentissage, il faut sélectionner un sous-ensemble de ces descripteurs pour éviter le problème du coût de traitement ainsi que le problème de faible fréquence de certains termes.

3. La reconnaissance de l'écriture : Dans le domaine de la reconnaissance de l'écriture manuscrite, les attributs peuvent être décrits comme un moyen de distinguer les entités appartenant à une classe (mots, lettres, chiffres) de celles appartenant à une autre classe (mots, lettres, chiffres). De plus, ils ne sont pas tous informatifs. Certains attributs sont sensibles au bruit, sans importance, corrélés, sans rapport avec la tâche en cours d'exécution, La sélection des attributs pertinents est donc devenue une étape importante dans tous les systèmes de reconnaissance de l'écriture manuscrite [26].

4. La reconnaissance d'images : Un des plus grands problèmes de la procédure de reconnaissance d'images avec des grandes tailles est toujours le «problème de la dimension ». La réduction de dimension est une approche prometteuse pour résoudre ce problème, et les algorithmes de sélection d`attributs sont souvent appliqués pour optimiser les performances de classification des systèmes de reconnaissance d'images [27].

5. La bioinformatique : Au cours de la dernière décennie, la motivation pour l`application des techniques de la sélection d`attributs dans le domaine de la bioinformatique a dépassé le fait d`être un exemple illustratif pour devenir une véritable condition préalable à la construction des modèles. Pour faire face à ces problèmes, plusieurs techniques de sélection d`attributs ont été conçues par des chercheurs en bioinformatique, en apprentissage automatique et en extraction de connaissances [28].

Conclusion:

La sélection des attributs est un problème majeur dans de nombreux domaines. Par conséquent, il a été un sujet d'intérêt pour de nombreux chercheurs. Dans ce chapitre, nous avons présenté l'importance et les avantages de la réduction de dimension tout en décrivant les deux approches de la réduction de dimension: l'extraction d'attributs et la sélection d'attributs. Après avoir présenté en détail la sélection des attributs et ses différents avantages, nous avons exposé le processus de sélection des attributs. Puis nous avons détaillé les différentes étapes de ce processus. Dans le chapitre suivant, nous proposons un état de l'art en présentant différentes méthodes de sélection des attributs de la littérature

Chapitre 2 :

Détection de maliciels Android

Introduction

Le nombre de menaces cachées sur les appareils mobiles augmentant rapidement, les développeurs sont conscients de l'importance de la sécurité des applications mobiles, mais celle-ci n'est pas largement comprise. Non seulement la fraude mobile est en augmentation, mais les institutions financières doivent prendre au sérieux la sécurité des applications mobiles et s'engager dans une stratégie globale. Les consommateurs doivent faire attention aux informations qu'ils publient et aux données qu'ils téléchargent lorsqu'ils surfent sur Internet. Pour analyser et détecter les applications malveillantes sur la plateforme Android, il est important de comprendre le fonctionnement des applications Android et d'étudier les différentes techniques de détection disponibles. Ce chapitre analyse la sécurité des applications mobiles Android, les vecteurs d'attaque utilisés par les auteurs de logiciels malveillants pour infecter les appareils mobiles et identifie diverses techniques d'analyse des applications Android.

I. La sécurité des applications mobiles :

Les appareils mobiles transmettent et reçoivent également des informations via Internet, à l'inverse d'un réseau privé. Cela les rend vulnérables aux attaques. Les entreprises peuvent tirer parti des réseaux privés virtuels (VPN) pour ajouter une couche de sécurité des applications mobiles pour les collaborateurs qui se connectent aux applications à distance. Les départements informatiques peuvent également décider de valider les applications mobiles, en s'assurant qu'elles sont conformes aux stratégies de sécurité de la société avant d'autoriser les employés à utiliser les applications mobiles connectées au réseau d'entreprise [29].

II. Les caractéristiques de sécurité mobiles Android :

- **Force:** L'une des fonctionnalités de sécurité les plus importantes d'Android est le contrôle des autorisations. Les applications doivent obtenir une autorisation avant d'utiliser des données ou des fonctionnalités mobiles. Les utilisateurs peuvent consulter les autorisations accordées à chaque application et les modifier à tout moment [30].
- **Contrôle**: Android fournit une fonctionnalité appelée "APP Sand box[2] " qui aide à isoler les applications les unes des autres. En d`autres termes, si l'application plante, vous ne pourrez pas accéder à vos données ni à d'autres applications.
- **Protection contre les logiciels malveillants** : Google Play Protect est le système de sécurité intégré d'Android qui vérifie la sécurité des applications avant leur installation. Si l'application est classée comme malveillante, elle sera bloquée

[2] Sand box : est une fonctionnalité de sécurité qui empêche Access d'exécution de certaines expressions potentiellement dangereuses. Ces expressions non sûres sont bloquées, que la base de données soit « fiable » (son contenu est activé).

- **Mise à jour de sécurité** : Android publie régulièrement des mises à jour de sécurité qui corrigent les vulnérabilités de sécurité connues. Pour bénéficier de ces mises à jour, il est important de maintenir votre téléphone à jour [30].
- **Écran verrouillé**: Le verrouillage de l'écran permet d'empêcher tout accès non autorisé à votre téléphone en cas de perte ou de vol de votre téléphone. Android propose diverses options de verrouillage d'écran telles que le code PIN, le mot de passe, le motif, le visage et l'empreinte digitale.
- **Développement de produits** : Android dispose d'une fonction de cryptage des données qui protège les données stockées sur votre téléphone. Cela signifie que toute personne ayant accès à votre téléphone ne peut pas accéder à vos données sans le mot de passe de cryptage.
- **VPN intégré** : Android est livré avec un VPN intégré qui sécurise votre connexion Internet lorsque vous vous connectez à des réseaux publics ou non sécurisés.

Ces fonctionnalités de sécurité pour Smartphone Android vous aident à vous protéger, vous et votre Smartphone, contre les attaques malveillantes [30]

III. les autorisations des applications Android :

Les autorisations d'applications Android sont des demandes d'accès aux fonctionnalités et aux données de l'appareil. Lorsque vous installez une application à partir du Google Play Store ou d'autres sources, elle vous demande généralement l'autorisation d'accéder à certaines données ou fonctionnalités sur votre téléphone. Cela inclut les caméras, les contacts, les lieux, les fichiers, etc. Les autorisations d'application sont importantes car elles permettent aux utilisateurs de contrôler les données et les fonctionnalités auxquelles une application peut accéder [31]. Si une application ne demande pas l'autorisation d'accéder à une fonctionnalité ou à des données particulières, elle ne pourra pas accéder à cette fonctionnalité ou à ces données.

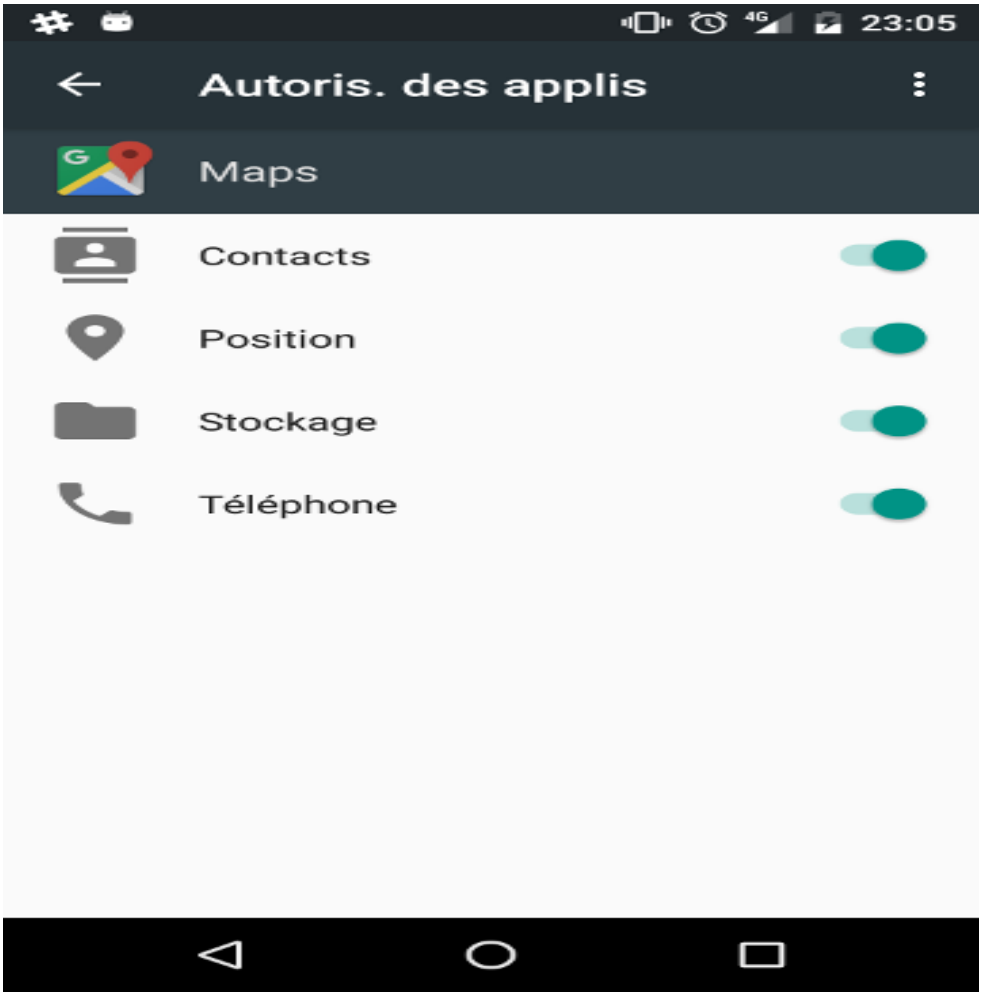


Figure2.1 : Exemple des Permissions pour l'application Maps

III.1 Différents types Permissions :

Group de permission	Autorisation
Agenda	Ajouter ou modifier des évènements d'agenda et envoyer Des e-mails aux invités à l'insu du propriétaire
Téléphones	- Lire le journal d'appels - Modifier le journal d'appels - Ajouter des messages vocaux
Contacts	-Lire vos contacts - Modifier vos contacts
Stockage	-Lire le contenu de la carte SD - Modifier ou supprimer le contenu de la carte SD
Localisation	-Accéder à votre position précise - Accéder à votre position approximative
Capteurs Corporels	Accéder aux capteurs corporels (comme le moniteur de Fréquence cardiaque)
Appareil Photo	Prendre des photos et filmer des vidéos

Tableau 2.1 : Permissions dangereuses et leur groupe sur Android [33].

IV. Modèle de sécurité Android :

Le modèle de sécurité d'Android a très peu changé depuis sa création et a été largement examiné dans [34]. Néanmoins, il reste une exigence obligatoire. Nous avons effectué une analyse complète des risques sur la plateforme.

1. Signature numérique

Cela confond même les informaticiens qui associent couramment l'existence de certificats à l'inviolabilité de la sécurité, peut-être à cause de la propagande du commerce électronique. Quelque chose à propos du protocole HTTPS (le fameux avantage du cadenas jaune et de nos jours la barre d'adresse verte).En fait, tous les systèmes de signature numérique apparus récemment sur le marché grand public (par exemple Pour éviter les problèmes d'expiration et de renouvellement des clés, Google exige la signature de certificats dont la date d'expiration est supérieure [34].

2. Cloisonnement :

Chaque application se voit attribuer un compte lors de l'installation Unix (UID). L'isolation entre les applications est assurée par les mécanismes suivants : Sécurité native pour les systèmes Unix. Toutes les applications signées avec le même certificat s'exécutent sous le même ID de groupe Unix (GID). Les applications signées avec le même certificat sont très susceptibles d'interagir {permettant aux applications d'utiliser le même UID[3] , ouvrant la porte à la création d'innombrables applications malveillantes. Chaque application a un ensemble limité d'autorisations (individuellement, elles ne sont pas considérées comme très dangereuses), mais combiner toutes ces applications dans le même processus augmenterait toutes les données sur un téléphone mobile.

3. Révocation :

En raison de la prolifération croissante d'applications malveillantes, Google utilise de plus en plus un mécanisme appelé kill Switch[4] dans son MarketPlace officiel. Ce mécanisme permet à Google de supprimer à distance toutes les instances d'une application identifiée par un certificat [35,36]. Le fonctionnement de ce mécanisme a été largement analysé et repose sur le processus Un GTalkService, N'importe qui disposant de certificats valides SSL peut émettre de tels messages, mais la liste des autorités approuvées se trouve dans le fichier /system/etc./Security/cacerts.bks, ce qui rend difficile leur modification. Cependant, une application malveillante pouvant élever les privilèges root peut contourner cette sécurité. L'activation du mécanisme du kill Switch bloque l'utilisation du kill Switch sur certains appareils, mais cela repose sur des hypothèses telles que la connexion de l'appareil cible à votre réseau de données et sa configuration pour en tirer parti. La sécurité basée sur la révocation a montré ses limites dans le passé, ce qui souligne les défis de ce mécanisme

V. Menaces de sécurité des applications mobiles :

Les applications mobiles peuvent présenter plusieurs menaces de sécurité, voici quelques-unes des plus courantes [37] :

- **Malwares** : Les malwares, tels que les virus, les chevaux de Troie et les logiciels espions, peuvent être intégrés dans des applications mobiles malveillantes. Ils peuvent compromettre la confidentialité des données de l'utilisateur, voler des informations sensibles ou causer des dommages au téléphone.

- **Fuites de données** : Les applications mal sécurisées peuvent exposer les données personnelles des utilisateurs en raison de vulnérabilités ou d'erreurs de développement. Cela peut inclure des informations telles que les noms d'utilisateur, les mots de passe, les adresses e-mail, les numéros de téléphone et même les données financières.

[3] http://developer.android.com/reference/android/R.attr.html#sharedUserId
[4] http://android-developers.blogspot.com/2010/06/exercising-our-remote-application.html

- **Man-in-the-Middle (MITM)** : Les attaques MITM se produisent lorsque les communications entre l'application mobile et le serveur sont interceptées par un attaquant. Cela peut permettre à l'attaquant d'intercepter et de lire les données sensibles échangées, telles que les informations d'identification, les données financières ou les messages privés [37].

- **Décompilation** : Les applications mobiles peuvent être décompilées, ce qui signifie que le code source de l'application peut être extrait. Cela peut faciliter l'analyse de l'application par des pirates informatiques, qui peuvent trouver des vulnérabilités et développer des attaques ciblées.

- **Contrefaçon d'application** : Les applications malveillantes peuvent imiter des applications légitimes pour tromper les utilisateurs et les inciter à fournir des informations sensibles. Ces applications contrefaites peuvent être distribuées via des applications stores tiers ou des liens de téléchargement malveillants.

- **Autorisations excessives** : Certaines applications demandent des autorisations excessives lors de l'installation, ce qui signifie qu'elles peuvent accéder à des données ou des fonctionnalités du téléphone qui ne sont pas nécessaires pour leur bon fonctionnement. Cela peut exposer les utilisateurs à des risques de sécurité et de confidentialité [37].

Pour se protéger contre ces menaces, il est recommandé de prendre les mesures

Suivantes [37] : Télécharger des applications uniquement à partir de sources fiables, telles que les applications stores officiels.

- ✓ Lire les avis et les évaluations des utilisateurs avant de télécharger une application.
- ✓ Vérifier les autorisations demandées par une application et être attentif aux autorisations excessives.
- ✓ Maintenir le système d'exploitation et les applications à jour avec les dernières mises à jour de sécurité.
- ✓ Utiliser des solutions de sécurité mobile, telles que des applications antivirus et des pare-feu.
- ✓ Éviter de se connecter à des réseaux Wifi publics non sécurisés lors de l'utilisation d'applications qui nécessitent l'envoi d'informations sensibles.
- ✓ Être vigilant quant aux signes d'activité suspecte ou de comportement anormal de l'application et signaler tout problème aux développeurs ou aux fournisseurs d'applications.

VI. Les techniques de détection de maliciel sur Android :

VI.1. Analyse statique :

L'analyse statique des logiciels malveillants est une méthode qui permet d'examiner un échantillon de logiciel malveillant sans réellement l'exécuter [38]. Elle fournit une estimation approximative de la manière dont le logiciel malveillant peut affecter l'environnement lors de son exécution, sans pour autant l'exécuter réellement. L'avantage principal de l'analyse statique

est sa capacité à découvrir tous les scénarios comportementaux possibles et à permettre aux chercheurs de voir toutes les façons dont un logiciel malveillant peut s'exécuter. De plus, ce type d'analyse est plus sûr que l'analyse dynamique, car le fichier n'est pas exécuté, ce qui évite les conséquences néfastes sur le système. Cependant, l'inconvénient majeur de l'analyse statique est le temps nécessaire pour effectuer cette analyse. En conséquence, elle n'est généralement pas utilisée dans des environnements dynamiques du monde réel, tels que les systèmes antivirus. Elle est souvent utilisée à des fins de recherche, par exemple lors du développement de signatures pour les logiciels malveillants. L'analyse statique peut comprendre différentes techniques [39]:

1. **Inspection du format de fichier :** Les métadonnées de fichier fournissent des informations utiles. Par exemple, Le fichier Windows PE (Portable Exécutable) fournit de nombreuses informations sur les fichiers Windows PE. Temps de compilation, fonctions importées et exportées, etc.
2. **Extraction de la chaîne de caractères :**Il s'agit de l'examinassions de la sortie du logiciel (messages d'état ou d'erreur) et l'inférence d'information sur l'opération du programme malveillant.
3. **Empreintes digitales :** Cela inclut le calcul des hachages cryptographiques, la recherche d'artefacts environnementaux, Codes de nom d'utilisateur, noms de fichiers, chaînes de registre, etc.
4. **Analyse d'antivirus :** Si les fichiers vérifiés sont des logiciels malveillants connus, probablement tous les scanners. Les programmes antivirus peuvent le détecter. Bien que cela puisse sembler hors de propos Cette méthode de détection est couramment utilisée par les vendeurs d'équipements audiovisuels. Ou vérifiez le résultat dans le bac à sable**.**
5. **Démontage** : La méthode d'analyse statique la plus courante et fiable consiste à désassembler le code machine pour le traduire en langage d'assemblage, puis à déduire la logique et les intentions du logiciel à partir de cette traduction.[39]

VI.2. Analyse dynamique :

Contrairement à l'analyse statique, cette méthode consiste à surveiller le comportement d'un fichier pendant son exécution et à déduire ses propriétés à partir de ces informations. En utilisant cette technique, nous pouvons observer toutes les fonctionnalités du logiciel malveillant et son impact sur l'environnement lorsqu'il est en cours d'exécution. Habituellement, le fichier est exécuté dans un environnement virtuel tel qu'un bac à sable (Sand box) [39]. Le bac à sable est un mécanisme de sécurité qui permet d'exécuter des programmes non approuvés dans un environnement isolé sans risque de nuire aux systèmes réels [38]. Grâce à cette analyse, il est possible de découvrir tous les comportements associés, tels que les fichiers ouverts, les mutex créés, etc., et elle est également plus rapide que l'analyse statique [39]. Cependant, cette approche présente deux inconvénients. D'une part, une seule séquence d'exécution ne représente pas l'ensemble du code, et une analyse négative ne garantit pas que le code soit sûr. D'autre

part, en se concentrant uniquement sur le comportement passé, cette approche est souvent insuffisante pour détecter rapidement une activité malveillante [40].

VII. Méthodes de détection de maliciel sur Android :

Les méthodes de détection des logiciels malveillants sont largement classées Ils entrent dans différentes catégories selon différentes perspectives. Voici comment **Figure 1.4** :

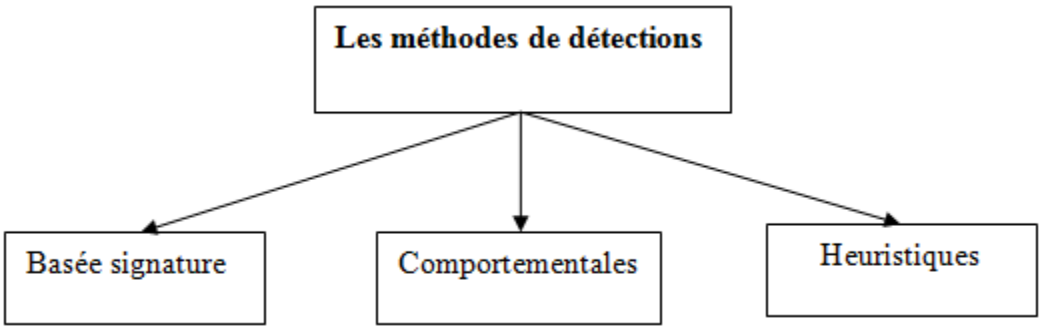


Figure 2.2: Méthodes de détection de malware [41].

VII.1. Méthodes basées signature :

Les programmes antivirus commerciaux utilisent largement ces méthodes. Elles reposent sur la représentation de chaque logiciel malveillant à l'aide d'une signature. Cette signature est une séquence d'octets unique pour chaque fichier malveillant, similaire à une empreinte digitale d'un exécutable. Par exemple, cela peut être un hachage MD5 ou SHA1, des chaînes statiques ou des métadonnées de fichier [39] [41].Prenons l'exemple du malware Chernobyl/CIH[5], qui possè de la signature suivante : E8000000005B8D4B425150500F014C24FE5B83C31CFA8B2B [42]. Après leur extraction, les signatures sont stockées dans ce qu'on appelle une base virale. À chaque fois qu'un fichier est analysé, sa signature est générée et comparée à toutes les autres présentes dans cette base. Les méthodes basées sur les signatures utilisent ces modèles extraits des divers malwares pour les identifier, ce qui les rend plus efficaces et plus rapides que d'autres méthodes. Cependant, elles présentent certains inconvénients. Elles sont incapables de détecter les variantes de logiciels malveillants inconnues et nécessitent beaucoup de temps et d'argent pour extraire les signatures uniques. De plus, elles ne sont pas efficaces contre les malwares qui modifient leur code à chaque infection, tels que les malwares polymorphes et métamorphiques [41].

Synthèse : Ces techniques utilisent des motifs spécifiques pour identifier les malwares connus, mais peut être contournée par des malwares polymorphes ou inconnus. Les travaux dans ce

[5] Chernobyl qui Connu aussi sous le nom de Tchernobyl est un virus informatique créé pour infecter Les systèmes d'exploitation Microsoft Windows. CIH doit son nom initial à son inventeur taïwanais Cheng Ing-Hau. Il a été détecte pour la première fois en juin 1998.

domaine se concentrent sur la création de bases de données de signatures et sur des méthodes efficaces pour les comparer avec les fichiers suspects.

VII.2. Les méthodes heuristiques :

Ces méthodes visent à trouver des caractéristiques comportementales et/ou structurelles spécifiques aux fichiers malveillants, permettant ainsi de les distinguer des fichiers légitimes [43]. Dans cette approche, une valeur est attribuée à chaque fonctionnalité associée à un logiciel malveillant. Voici quelques exemples de fonctionnalités de logiciels malveillants connus [44]:

- Enregistrement des frappes clavier.
- Écriture dans des fichiers exécutables.
- Suppression de nouvelles clés de registre à des emplacements spécifiques dans le registre Windows.
- Suppression de fichiers sur le disque dur.
- Activité réseau suspecte.

Afin d'automatiser ce type de méthodes, les experts en sécurité informatique ont utilisé des techniques d'apprentissage automatique (machine Learning), plus précisément l'apprentissage supervisé, pour construire un modèle de classificateur capable de distinguer les logiciels malveillants des fichiers légitimes [41].

Synthèse : ces techniques Repose sur des règles et des heuristiques pour détecter les comportements malveillants, mais peut générer des faux positifs. Les travaux dans ce domaine se concentrent sur le développement de techniques pour détecter des activités suspectes, telles que l'autoréplication, la modification des fichiers système.

VII.3. Méthode comportementale :

Cette méthode de détection des logiciels malveillants repose sur l'analyse du comportement d'un programme afin de déterminer s'il est malveillant ou non. Contrairement aux techniques basées sur les signatures de fichiers, qui se concentrent sur la structure interne des fichiers, les techniques basées sur le comportement observent les actions effectuées par un fichier exécutable. Un détecteur basé sur le comportement se compose principalement des éléments suivants [45]:

1. **Collecteur de données**: il recueille des informations de manière statique ou dynamique.
2. **Interprète**: il analyse les données collectées lors de la phase précédente afin d'extraire celles jugées les plus pertinentes.
3. **Comparateur:** il est utilisé pour confronter cette représentation comportementale aux signatures de comportement

Synthèse : ces techniques Analyse le comportement des logiciels pour détecter les malwares, mais peut nécessiter une analyse approfondie et générer des faux positifs. Les travaux dans ce domaine se concentrent sur la surveillance des actions et des interactions d'un logiciel avec le système et sur la détection des comportements malveillants, tels que l'accès non autorisé aux fichiers, les tentatives de communication suspectes

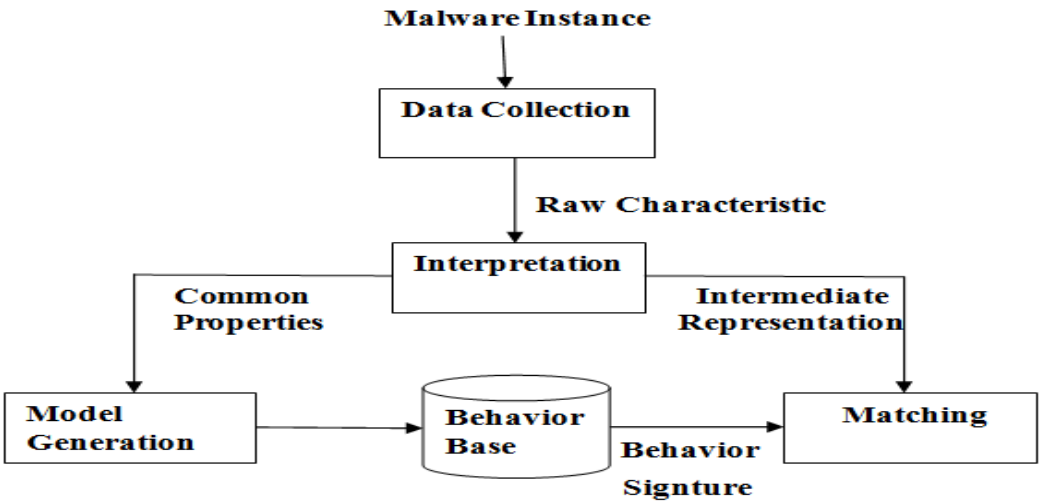


Figure 2.3 : Conception fonctionnelle d'un détecteur comportementale [45].

VIII. Les méthodes de data mining dans la détection de logiciels malveillants :

Le data mining est utilisé pour détecter les logiciels malveillants sur Android en analysant les caractéristiques et les comportements des applications afin d'identifier celles qui sont malveillantes. Les méthodes couramment utilisées [46] :

1 .Extraction de caractéristiques : Les techniques d'extraction de fonctionnalités permettent de capturer les attributs et propriétés des applications Android, ce qui aide à différencier les

applications malveillantes des applications légitimes. Elles fournissent des informations sur les autorisations, les actions, les composants et les interactions réseau des applications [46].

2. **Apprentissage supervisé** : L'utilisation de l'apprentissage supervisé implique la construction d'un modèle à partir d'un ensemble d'applications préalablement étiquetées comme malveillantes ou légitimes. Les caractéristiques extraites de ces applications sont utilisées pour entraîner le modèle, qui est ensuite en mesure de prédire la classe (malveillante ou légitime) des nouvelles applications.

3. **Apprentissage non supervisé :** L'utilisation de l'apprentissage non supervisé, comme le clustering, permet de regrouper des applications similaires en fonction de leurs caractéristiques extraites. Ce regroupement aide à identifier des ensembles d'applications qui partagent des comportements similaires ou suspects, ce qui peut mettre en évidence la présence de logiciels malveillants.

4. **Analyse de séquences :** est utilisée pour étudier les comportements des applications Android sur une période donnée. Elle consiste à analyser les séquences d'actions, les requêtes réseau, les interactions avec les composants du système et d'autres types de données. Cette approche permet de détecter les schémas de comportement malveillant en identifiant des séquences d'événements suspects ou anormaux [46]. Les modèles d'analyse de séquences sont utilisés pour repérer ces schémas et contribuent ainsi à la détection de logiciels malveillants sur Android.

5. **Réseaux de neurones :** Les réseaux de neurones, y compris les réseaux neuronaux profonds, permettent d'apprendre des représentations avancées des applications Android. Ces représentations sont capables de saisir des informations complexes sur les applications et peuvent être utilisées dans la détection de logiciels malveillants. Grâce à ces représentations apprises, il est possible d'identifier les applications malveillantes en exploitant les caractéristiques significatives extraites par les réseaux de neurones.

6. **Analyse de texte :** L'analyse de texte peut être appliquée aux descriptions d'applications et aux commentaires des utilisateurs afin d'extraire des informations pertinentes. Cette méthode permet notamment de repérer des termes ou des phrases suspectes qui peuvent être associés à des logiciels malveillants. En examinant le contenu textuel, on peut identifier des éléments qui soulèvent des soupçons et qui peuvent être indicatifs d'un comportement malveillant [46].

IX. Les travaux existants:

Dans cette section, nous examinerons quelques études similaires portant sur l'analyse et la détection des logiciels malveillants.

- **Gavrilut et al:** Une étude a développé un système de classification en temps réel pour détecter les logiciels malveillants. Les chercheurs ont utilisé 308 caractéristiques pour représenter ces logiciels et ont réduit leur nombre pour faciliter l'apprentissage. Ils ont utilisé un algorithme d'apprentissage automatique supervisé en ligne pour classifier les

fichiers comme bénins ou malveillants. Le modèle s'adapte aux changements et nouvelles menaces en continu, mais nécessite un flux constant de données et des ressources de calcul plus élevées que les techniques hors ligne traditionnelles [47].

- **Tony Abou-Assaleh et al :** Des chercheurs ont développé un modèle de détection de programmes malveillants en utilisant le classificateur KNN et la méthode d'analyse Common N-Gram. Ils ont utilisé une fenêtre glissante pour extraire des sous-chaînes d'octets chevauchantes à partir d'un fichier, puis ont généré des N-Gram fréquents à partir de ces sous-chaînes. Ces N-Gram ont servi de signatures pour prédire si un programme était malveillant ou inoffensif en le comparant à des échantillons connus. Lors de tests avec des exécutables Windows bénins et des vers provenant d'emails infectés, le modèle a atteint une précision moyenne de 98% lors de la Main Accuracy. Bien que cette méthode permette de détecter des programmes invisibles, les créateurs de virus pourraient tenter de la contourner [48].
- **Schultz et al :** Les auteurs ont développé le premier système de détection de logiciels malveillants en utilisant des techniques d'apprentissage automatique. Ils ont examiné diverses informations présentes dans les fichiers PE, telles que les chaînes de caractères, les API et les séquences d'octets. Pour la classification, ils ont utilisé un algorithme bayésien naïf, et ont obtenu une précision globale de 97,11% en se basant sur les chaînes de caractères comme attributs [49].
- **Zhenlong Yuan :** Une méthode AA a été proposée, qui extrait plus de 200 fonctions à la fois de l'analyse statique et de l'analyse dynamique des applications Android, pour détecter les logiciels malveillants. En comparant les résultats de modélisation, il a été démontré que la technologie de Deep Learning est particulièrement adaptée à la reconnaissance des logiciels malveillants Android, en utilisant de véritables ensembles d'applications Android [50].
- **Justin Sahs :** il présenté récente, un système novateur a été présenté, utilisant l'apprentissage automatique pour détecter les applications malveillantes sur les appareils Android. Ce système extrait diverses caractéristiques des applications, puis entraîne un algorithme SVM en dehors des appareils, sur des serveurs ou un cluster de serveurs. Cette approche permet d'exploiter une puissance de calcul plus élevée afin d'améliorer la détection des applications malveillantes [51].
- **Asaf Shabtai :** Il est présenté un cadre visant à détecter les applications malveillantes sur les appareils mobiles Android. Ce cadre propose un système de détection de logiciels malveillants basé sur l'hôte, qui surveille en permanence diverses fonctionnalités et événements obtenus à partir de l'appareil mobile. Ensuite, il utilise des détecteurs d'anomalies d'apprentissage automatique pour classifier les données collectées comme étant normales (bénignes) ou anormales (malveillantes)[52].
- **Naser Peiravian :** Il a proposé de fusionner les autorisations et les appels d'API, ainsi que d'utiliser des techniques d'apprentissage automatique pour détecter les applications Android malveillantes [53].
- **Karl Pearson** : Un statisticien britannique qui a introduit le concept d'analyse en composantes principales (ACP) en 1901, l'une des techniques les plus fondamentales pour la réduction de dimension. [4].

- **Thomas Cover et Joy Thomas** : Ils ont développé la méthode de sélection de variables basée sur l'information mutuelle (méthode d'information mutuelle maximale) largement utilisée pour la sélection d'attributs dans la réduction de dimension. [3]
- **Shaikh Bushra Almin :** Proposé un système pour détecter et supprimer les logiciels malveillants présents sur les appareils Android des utilisateurs [54].
- **Cheng lin Li :** Il a présenté un nouveau classificateur fiable pour détecter les malwares Android. Ce classificateur repose sur l'architecture de la machine de factorisation et utilise l'extraction des caractéristiques des applications Android à partir des fichiers manifestes et du code source pour effectuer la détection de manière précise [55].
- **Hossein Fereidooni :** Il a suggéré la mise en place d'un système de détection des applications Android malveillantes qui utilise une analyse statique du comportement des applications. Ce système offre une couverture étendue des différents comportements de sécurité. Par ailleurs, il a développé un cadre de détection basé sur l'apprentissage automatique, qui assure une détection efficace tout en maintenant un taux de faux positifs acceptable, comme indiqué dans la référence mentionnée **[56].**

Conclusion :

En conclusion, la sécurité d'Android demeure une préoccupation constante en raison de la popularité des appareils fonctionnant sur cette plateforme. Malgré les efforts considérables déployés pour renforcer la sécurité, de nouveaux défis émergent régulièrement en raison de l'évolution des techniques d'attaque des cybercriminels. Pour garantir la sécurité des utilisateurs, il est essentiel de maintenir une collaboration étroite entre les chercheurs en sécurité, les développeurs d'applications et les fabricants de dispositifs Android. Cette collaboration permet d'identifier rapidement les vulnérabilités, de mettre en place des correctifs et de fournir des mises à jour de sécurité régulières. En outre, les initiatives de sécurité communautaires, les programmes de divulgation responsable et les récompenses pour les signalements de vulnérabilités encouragent l'implication de la communauté dans l'amélioration de la sécurité d'Android .En somme, la sécurité d'Android est un processus dynamique qui nécessite une attention continue afin de protéger efficacement les utilisateurs dans Android

Chapitre 3 :

Contribution et Implémentation

Introduction :

Dans ce chapitre, nous explorons le processus de sélection des caractéristiques de l'ensemble de données. La sélection des caractéristiques est une étape critique de l'apprentissage automatique, car elle nous permet d'identifier les variables les plus importantes de l'ensemble de données et d'éliminer celles qui ne sont pas pertinentes, bruyantes ou redondantes. Cela permet d'améliorer les performances et la généralisation des modèles construits sur l'ensemble des données.

L'ensemble de données contient des observations qui peuvent présenter des valeurs manquantes et une distribution entre les classes. Cependant, il y a beaucoup de variables dans l'ensemble de données, ce qui représente un défi pour la construction et l'interprétation des modèles. Nous évaluons donc les performances de plusieurs classificateurs, notamment d'Arbre de Décision, Forêt Aléatoire, Classificateur de Vecteur de Support, k plus proches voisins, à différents niveaux de sélection des caractéristiques.

Ce chapitre présente les résultats du processus de sélection des caractéristiques et les mesures de performance des classificateurs pour chaque niveau de sélection des caractéristiques. Les mesures comprennent l'exactitude, la précision, le rappel, le F1- mesure, ainsi que le le temps nécessaire en secondes.

Dans l'ensemble, ce chapitre fournit des informations précieuses sur le processus de sélection des caractéristiques et son impact sur les performances des modèles d'apprentissage automatique. Les résultats peuvent être utilisés pour guider les recherches futures et améliorer la précision et l'efficacité des modèles construits sur des ensembles de données android.

I. Ensembles de données (dataset) :

Est une ressource précieuse pour les chercheurs intéressés par l'étude de la détection des logiciels malveillants sur les appareils Android. Il fournit des informations complètes sur les différents types d'autorisations associées aux applications Android et peut être utilisé pour former des algorithmes d'apprentissage automatique visant à détecter les applications malveillantes.

Parmi les variables les plus importantes de l'ensemble de données figurent les suivantes :

- Permissions de l'application : Comme indiqué précédemment, cette variable représente les différentes autorisations demandées par chaque application Android.
- Type d'application : Cette variable indique si une application est un goodware ou un maliciel .
- Nom du paquet : cette variable représente le nom unique attribué à chaque paquet d'applications Android.
- Taille du fichier : Taille du fichier d'installation de l'application.
- Version minimale du SDK : Version minimale d'Android requise pour que l'application fonctionne.
- Version SDK cible : La version d'Android pour laquelle l'application a été développée.

II.1. Les ensembles de données (dataset) utilisés

1. **TUANDROMD** (Tezpur University Android Malware Dataset) [57]: Est un ensemble de données complet qui fournit des informations précieuses pour la recherche sur la détection des logiciels malveillants sur les appareils Android. L'ensemble de données comprend 4465 observations et 242 variables, chacune d'entre elles correspondant à une application Android particulière. Les applications de l'ensemble de données sont étiquetées comme "goodware" ou "malware", les applications "goodware" prenant la valeur "0" et les applications "malware" la valeur "1". L'ensemble de données a été créé par orah, Parthajit, D. K. Bhattacharyya et J. K. Kalita dans le cadre de leurs recherches sur la génération et l'évaluation d'ensembles de données sur les logiciels malveillants.

Distribution de classe	Nombre d'observations	Nombre De variables	Valeurs manquantes	Cellules manquantes (%)	Taille totale en mémoire
malware:3564 goodware: 899	4465	242	0	0	8.5 MB

Tableau 3.1: Statistiques TUANDROMD

2. **GENOME dataset** [58]: le jeu de données combiné proviennent du jeu de données Genome APK. Les auteurs ont collecté plus de 1 200 échantillons de logiciels malveillants. Ces échantillons ont été collectés afin de couvrir un large éventail de familles de logiciels malveillants Android, allant de leurs premières apparitions en août 2010 aux plus récentes en octobre 2011. Ils ont été caractérisés en détail, incluant leurs méthodes d'installation, leurs mécanismes d'activation et la nature de leurs charges malveillantes. La caractérisation et l'étude basée sur l'évolution de familles représentatives ont démontré leur évolution rapide pour échapper à la détection par les logiciels antivirus mobiles existants. Les fichiers "benign.csv" et "malign.csv" issus du jeu de données Genome APK ont été utilisés pour créer le jeu de données combiné, offrant aux chercheurs une vaste collection d'applications Android bénignes et malveillantes pour étudier et développer des techniques de détection robustes ainsi que des stratégies de mitigation. par la détection de logiciels malveillants sur Android. Sa vaste gamme de fonctionnalités et sa grande taille d'échantillon en font un outil précieux pour développer et tester de nouvelles méthodes et techniques de détection.

Distribution de classe	Nombre d'observations	Nombre De variables	Valeurs manquantes	Cellules manquantes (%)	Taille totale en mémoire
0: 500 1: 500	1000	4066	0	0	31 MB

Tableau 3.2: Statistiques Genome

II. Techniques de sélection :

Dans cette section, nous allons discuter des techniques de sélection de fonctionnalités utilisées dans notre analyse. Les deux techniques utilisées sont l'information mutuelle et le test du chi- Square.

II.1. Information mutuelle (mutual information) :

L'information mutuelle d'un couple (X,Y) de variables représente leur degré de dépendance au sens probabiliste. Ce concept de dépendance logique ne doit pas être confondu avec celui de causalité physique, bien qu'en pratique l'un implique souvent l'autre. Informellement, on dit que deux variables sont indépendantes si la réalisation de l'une n'apporte aucune information sur la réalisation de l'autre. La corrélation est un cas particulier de dépendance dans lequel la relation entre les deux variables est strictement monotone. L'information mutuelle est nulle ssi les variables sont indépendantes, et croit lorsque la dépendance augmente.

Définition

Soit (X,Y) un couple de variables aléatoires de densité de probabilité jointe données par P(x,y). Notons les distributions marginales P(x) et P(y). Alors l'information mutuelle est dans le cas discret:

$$I(X,Y) = \sum_{x,y} P(x,y) \times \log \frac{P(x,y)}{P(x)P(y)} \quad \text{.....(1)}$$

et, dans le cas continu:

$$I(X,Y) = \int_{(-\infty,\infty)\times(-\infty,\infty)} P(x,y) \times log \frac{P(x,y)}{P(x)P(y)}\, dxdy \text{.....(2)}$$

Où P(X,Y), P(X) et P(Y) sont respectivement les densités des lois de (X,Y), X et Y [59]L'information mutuelle mesure la quantité moyenne d'informations que les variables X et Y partagent. Elle est calculée en comparant la distribution jointe P(x, y) avec le produit des distributions marginales P(x) et P(y). Une information mutuelle nulle indique que les variables X et Y sont indépendants, c'est-à-dire que la réalisation de l'une n'apporte aucune information sur la réalisation de l'autre. Plus l'information mutuelle est élevée, plus les variables X et Y sont dépendantes.

Il est important de noter que l'information mutuelle ne permet pas de déterminer la relation de causalité entre les variables. Bien qu'une forte dépendance probabiliste puisse suggérer une relation causale, ce n'est pas une certitude. La corrélation, qui est un cas particulier de dépendance où la relation entre les variables est strictement monotone, peut également être mesurée à l'aide de l'information mutuelle.

En résumé, l'information mutuelle quantifie la dépendance entre deux variables aléatoires, en mesurant la quantité d'informations qu'elles partagent. Elle est nulle lorsque les variables sont indépendantes et augmente à mesure que la dépendance entre elles augmente.

II.2. Chi-Square :

Le test du chi-Square est un test statistique utilisé pour déterminer si deux variables catégorielles sont indépendantes ou non. Il calcule la différence entre la fréquence observée et la fréquence attendue de chaque catégorie et détermine la probabilité que la distribution observée se produise par hasard. Dans le contexte de la sélection de fonctionnalités, le test du chi-Square mesure la dépendance entre chaque fonctionnalité et la variable cible. Plus la valeur du test du chi-square est élevée, plus la fonctionnalité dépend de la variable cible.

Nous avons choisi le test du chi-square car c'est une technique de sélection de fonctionnalités simple et efficace qui fonctionne bien avec les variables catégorielles. Elle est également capable de gérer des ensembles de données volumineux avec de nombreuses fonctionnalités.

Le prince de cette réduction :

$$x^2 = \sum_{j=0}^{j} \frac{(O_j - E_j)^2}{E_j} \text{..............................(4)}$$

Où
O_j = fréquence observé (valeur expérimentale)
E_j = fréquence envisagé (valeurs attendue)
j = Numéro de la catégorie de la valeur
J= Nombre total de catégorie de valeur » [60].

II.3.Valeur F de l'ANOVA :

La valeur F de l'ANOVA est une mesure statistique super utile pour tester s'il y a une différence importante entre les moyennes de deux ou plusieurs groupes. C'est un peu comme comparer les notes moyennes de deux classes différentes pour voir si l'une est vraiment meilleure que l'autre. Si la valeur F est grande, ça veut dire qu'il y a une forte chance qu'au moins un groupe ait une moyenne différente des autres.

Dans la sélection de caractéristiques, la valeur F de l'ANOVA est utilisée pour voir si une caractéristique est significativement liée à la variable cible dans un problème de classification. La fonction

La formule pour calculer la valeur F dans une ANOVA à un facteur est la suivante:

$$F = SSB \,/\, ddlB \div SSW \,/\, ddlW \text{..............................(5)}$$

où :
F représente la statistique F.

SSB : est la somme des squares entre les groupes.
ddlB : est le nombre de degrés de liberté entre les groupes.
SSW : est la somme des squares à l'intérieur des groupes.
ddlW : est le nombre de degrés de liberté à l'intérieur des groupes.

Pour utiliser cette équation, vous devez calculer les valeurs respectives de SSB, ddlB, SSW et ddlW : en fonction de votre ensemble de données, puis les substituer dans l'équation pour obtenir la statistique F. La statistique F peut ensuite être comparée à la valeur critique de F pour déterminer la significativité statistique des différences entre les moyennes des attributs ou des groupes.

Il est important de se rappeler que la valeur F de l'ANOVA a certaines hypothèses qui doivent être satisfaites pour que les résultats soient fiables. Si les données ne suivent pas une distribution normale ou si les échantillons ne sont pas également répartis entre les classes cibles, les résultats peuvent ne pas être précis. Dans ce cas, il peut être nécessaire d'utiliser d'autres méthodes de sélection de caractéristiques ou de prétraitement des données[61].

II.4. Élimination récursive des caractéristiques (RFE) :

La méthode de sélection de fonctionnalités Récursive avec élimination de caractéristiques (RFE) est une technique qui permet de sélectionner les variables les plus pertinentes pour un modèle en éliminant itérativement les variables les moins importantes. Elle est souvent utilisée en combinaison avec des modèles de régression ou de classification tels que la régression linéaire ou les SVM.

RFE classe les caractéristiques selon les attributs « coef » ou « importance des caractéristiques » du modèle. Il élimine ensuite de manière récursive un nombre mineur de fonctionnalités par boucle, supprimant toutes les dépendances et colinéarités existantes présentes dans le modèle

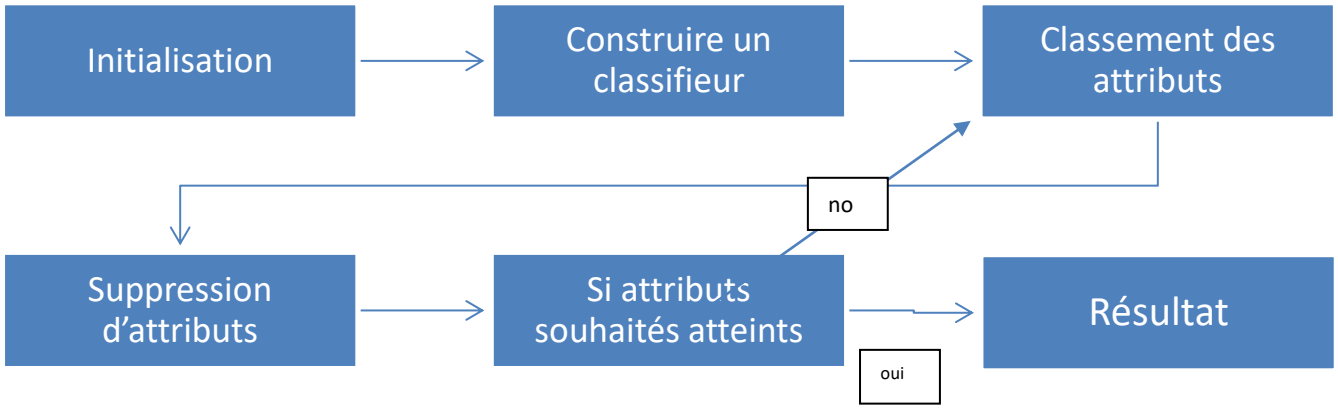


Figure 3.1 : structure de RFE

III. Langage et API utilisés :

- **Python** :

n’est pas un langage bon juste dans l’ingénierie des données. En dehors des Python n'est pas seulement un bon langage pour l'ingénierie des données. En dehors des bibliothèques telles que Panda ou NumPy, il intègre également une bibliothèque essentielle pour la science des données Scikit-Learn. De plus, il est également possible d'utiliser la bibliothèque PyQt5 pour développer des interfaces graphiques conviviales.

Voici la liste des principales bibliothèques utilisées dans le code :

- **Python binding for Qt (5PyQt5) :**

PyQt5 est un module qui vous permet de lier le langage Python à la bibliothèque Qt. Il vous permet de créer des interfaces graphiques en Python. Une extension de QtDesigner (utilitaire graphique pour créer des interfaces Qt) permet de gérer le code python des interfaces graphiques. PyQt possède tous les avantages du célèbre .

- **Pandas :**

La bibliothèque logicielle open-source Pandas est spécifiquement conçue pour la manipulation et l’analyse de données en langage Python. Elle est à la fois performante, flexible et simple d’utilisation.

Grâce à Pandas, le langage Python permet enfin de charger, d’aligner, de manipuler ou encore de fusionner des données. Les performances sont particulièrement impressionnantes quand le code source back-end est écrit en C ou en Python.

Le nom « Pandas » est en fait la contraction du terme « Panel Data » désignant les ensembles de données incluant des observations sur de multiples périodes temporelles. Cette bibliothèque a été créée comme un outil de haut niveau pour l’analyse en Python.

Les créateurs de Pandas comptent faire évoluer cette bibliothèque pour qu’elle devienne l’outil d’analyse et de manipulation de données open-source le plus puissant et flexible dans n’importe quel langage de programmation. [62]

- **scikit-learn:**

Scikit-Learn est une bibliothèque Python spécialisée dans dans les travaux de Data Science. C’est une bibliothèque facilement accessible, et puissante, qui s’intègre naturellement dans l’écosystème plus large des outils de science des données basés sur Python. Utilise

- **feature_selection** (sélection de caractéristiques) : Processus de sélection des caractéristiques les plus pertinentes ou informatives dans un jeu de données afin de réduire la dimensionnalité et d'améliorer les performances des modèles d'apprentissage automatique.
- **train_test_split** (division de l'ensemble de données en entraînement et test) : Une fonction de la bibliothèque scikit-learn (sklearn) qui permet de diviser un ensemble de données en deux

parties distinctes : un ensemble d'entraînement utilisé pour ajuster le modèle et un ensemble de test utilisé pour évaluer les performances du modèle.

- **cross_val_score** (Main Accuracy) : Une fonction de la bibliothèque scikit-learn (sklearn) qui permet d'évaluer les performances d'un modèle en utilisant la Main Accuracy. La Main Accuracy est une technique d'évaluation qui divise l'ensemble de données en plusieurs sous-ensembles, puis effectue plusieurs itérations d'entraînement et de test pour évaluer les performances du modèle de manière robuste.
- **DecisionTreeClassifier** (classifieur par arbre de décision) : Un algorithme d'apprentissage supervisé de la bibliothèque scikit-learn (sklearn) qui construit un arbre de décision à partir des données d'entraînement pour effectuer des classifications.
- **RandomForestClassifier** (classifieur de forêt aléatoire) : Un algorithme d'apprentissage supervisé de la bibliothèque scikit-learn (sklearn) qui construit un ensemble de plusieurs arbres de décision, puis utilise le vote majoritaire pour effectuer des classifications.
- **SVC** (Support Vector Classifier) : Un classifieur à vecteurs de support (SVC) de la bibliothèque scikit-learn (sklearn) qui effectue des classifications en utilisant des vecteurs de support dans un espace de grande dimension.
- **KNeighborsClassifier** (classifieur des k plus proches voisins) : Un algorithme d'apprentissage supervisé de la bibliothèque scikit-learn (sklearn) qui effectue des classifications en se basant sur les k exemples les plus proches dans l'espace des caractéristiques.
- **SelectPercentile** (sélection de pourcentage) : Une méthode de sélection de caractéristiques de la bibliothèque scikit-learn (sklearn) qui permet de sélectionner les meilleures caractéristiques en fonction d'un pourcentage donné.
- **mutual_info_classif** (information mutuelle pour les tâches de classification) : Une mesure de la dépendance statistique entre deux variables, utilisée dans la sélection de caractéristiques pour les tâches de classification.
- **chi2** (test du chi-deux) : Un test statistique utilisé pour évaluer la dépendance entre deux variables catégorielles, souvent utilisé dans la sélection de caractéristiques pour les tâches de classification.
- **f_classif** est une méthode de sélection de caractéristiques dans la bibliothèque sklearn qui calcule la valeur F de l'ANOVA pour un ensemble de données
- **RFE** est une technique d'élimination récursive des attributs qui vise à sélectionner les attributs en éliminant de manière récursive les attributs en fonction de leur moins grande importance qui leur est attribuée.
- **Dominate :**

  Dominate est une bibliothèque Python pour générer des pages HTML de manière programmatique. Elle offre une API simple pour créer des éléments HTML, du texte, des attributs et des éléments imbriqués.

  Dominate permet de générer des documents HTML complets, ce qui est utile pour :
  - Créer des pages HTML statiques
  - Générer des rapports au format HTML
  - Construire des applications web sans framework complet
  - Dominate définit les éléments HTML comme des classes Python.

  Caractéristiques clés :

- API Python simple
- Fermeture automatique des éléments
- Ajouter du texte et des éléments imbriqués
- Définir les attributs avec des mots-clés
- Bibliothèque d'éléments complète
- Balises intégrées pour CSS, JS et métadonnées

Dominate rend facile la génération d'HTML depuis Python. La bibliothèque est petite mais puissante pour générer de l'HTML.

IV. Expérimentation et Discussion

Dans l'analyse, plusieurs classificateurs ont été utilisés pour la détection de logiciels malveillants sur le jeu de données. Les classificateurs sont des algorithmes d'apprentissage automatique qui ont été entraînés sur le jeu de données pour prédire si une application Android est malveillante ou non..

Les classificateurs utilisés dans l'analyse sont :

1. Le Classificateur d'Arbre de Décision : Il s'agit d'un algorithme de classification qui utilise une structure en arbre pour prendre des décisions. Il divise l'ensemble de données en sous-ensembles plus petits en fonction des caractéristiques des données, jusqu'à ce que les feuilles de l'arbre contiennent des exemples de la même classe.
2. Le Classificateur de Forêt Aléatoire : Il s'agit d'un algorithme de classification qui utilise plusieurs arbres de décision pour prendre des décisions. Il crée un ensemble de modèles d'arbres de décision et les combine pour améliorer la précision de la prédiction.
3. Le Classificateur de Vecteur de Support : Il s'agit d'un algorithme de classification qui utilise un hyperplan pour séparer les données en deux classes. L'hyperplan est choisi de manière à maximiser la marge entre les deux classes.
4. Le Classificateur des k plus proches voisins : Il s'agit d'un algorithme de classification qui se base sur la similarité des exemples de données. Il choisit la classe d'un nouvel exemple en fonction de la classe des k exemples les plus proches dans l'ensemble de données.

Matrice de confusion : Une matrice de confusion, aussi appelée matrice d'erreur est une matrice N x N utilisée pour évaluer les performances d'un modèle de classification, où N est le nombre de classes cibles. La matrice compare les valeurs cibles réelles avec celles prédites par le modèle d'apprentissage automatique. Cela nous donne une vision globale de la performance de notre modèle de classification et des types d'erreurs qu'il commet. Elle comporte 4 valeurs essentielles :

		Prédit	
		Intrusion	NoIntrusion
Actuel	Intrusion	VP	FN
	NoIntrusion	FP	VN

- **Vrai positif (VP) :** Nombre de cas que le test déclare positifs et qui le sont réellement.
- **Faux positif (FP) :** Nombre de cas que le test déclare positifs et qui sont en réalité négatifs.
- **Vrai négatif (VN) :** Nombre de cas que le test déclare négatifs et qui sont en réalité négatifs.
- **Faux négatif (FN) :** Nombre de cas que le test déclare négatifs et qui sont en réalité positifs.

Nous avons plusieurs expérimentation sur 2 jeux de données : TUANDROMD et GENOME Pour chaque classificateur, les métriques suivantes sont signalées:

1. Exactitude (Accuracy): la fraction globale d'enregistrements classés correctement.

$$\text{Accuracy} = \frac{VP + VN}{VP + VN + FP + FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (6)$$

2. Précision: la fraction des vrais positifs (enregistrements correctement classifiés) parmi tous les enregistrements classifiés comme positifs par le modèle.

$$\text{Précision} = \frac{VP}{VP + FP} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (7)$$

3. Rappel: la fraction des vrais positifs parmi tous les enregistrements qui devraient avoir été classifiés comme positifs.

$$\text{Rappel} = \frac{VP}{VP + FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (8)$$

4. F1- mesure: la moyenne harmonique de précision et de rappel.

$$F1 = \frac{2 \times \text{Précision} \times \text{Rappel}}{\text{Précision} + \text{Rappel}} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (9)$$

5. Temps pris en secondes: le temps pris par le classificateur pour entraîner et évaluer le modèle.

IV.1. Ensemble de données TUANDROMD:

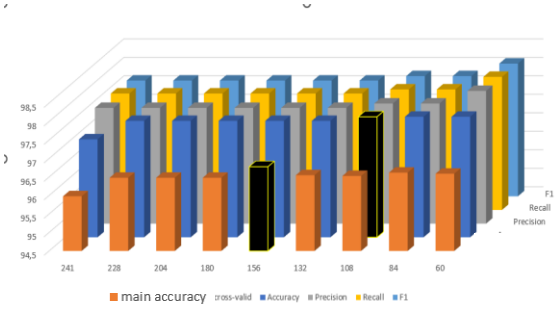
Le tableaux suivant montre la taille correspond au différentes sélections effectuées :

Nombre d'attributs	241	228	204	180	156	132	108	84	60
%	100	95	85	75	65	55	45	35	25
taille MB	8.21	7.76	6.95	6.13	5.31	4.49	3.68	2.86	2.04

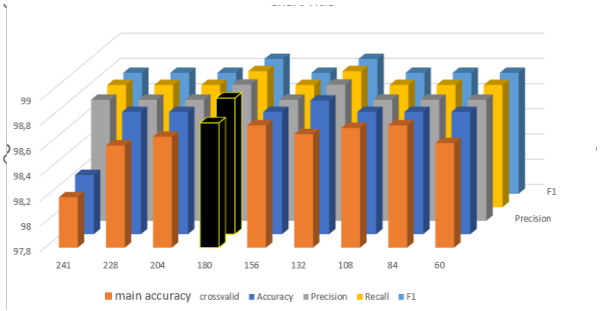
Tableau 3.3: les différentes sélections effectuées sur TUANDROMD

Après la sélection des attributs, les tailles du dataset peuvent être différentes en fonction des attributs conservés.

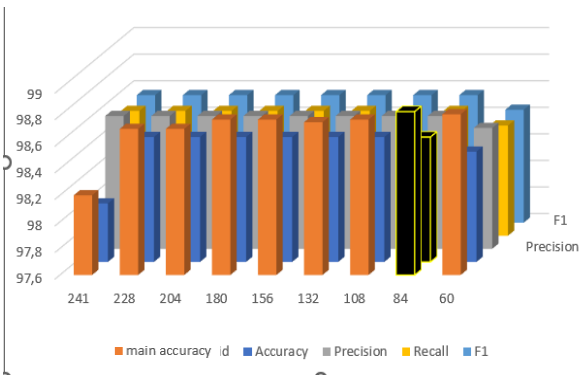
IV.1.1. Méthode 1 : RFE



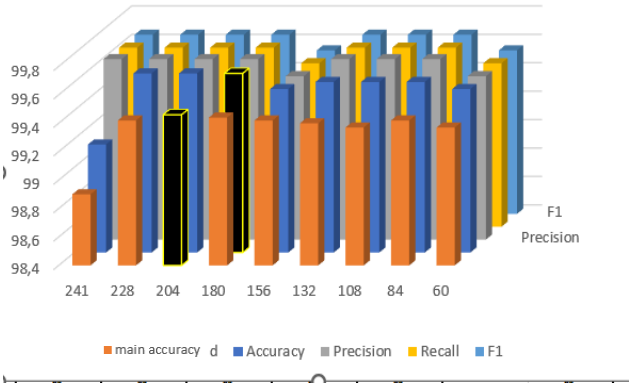
A1:k plus proches voisins



A2 :d'arbre de d écision



A3 : Vecteur de Support



A4: Forêt Aléatoire

Figure 3.2 : Résultat de la méthode RFE sur la base TUANDROMD

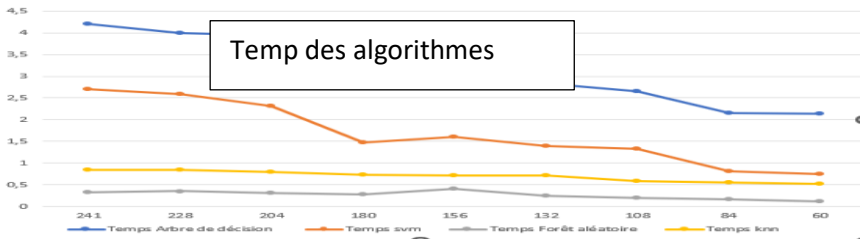
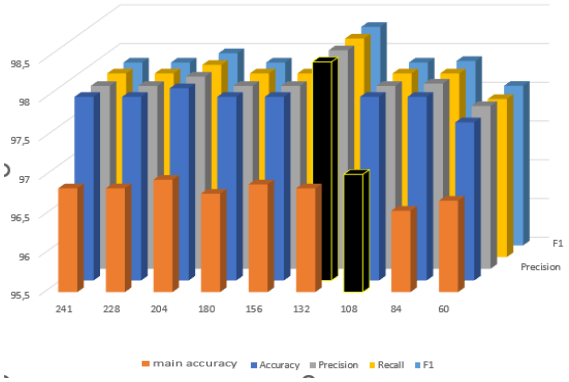


Figure 3.3 :temps des exécution des algorithmes

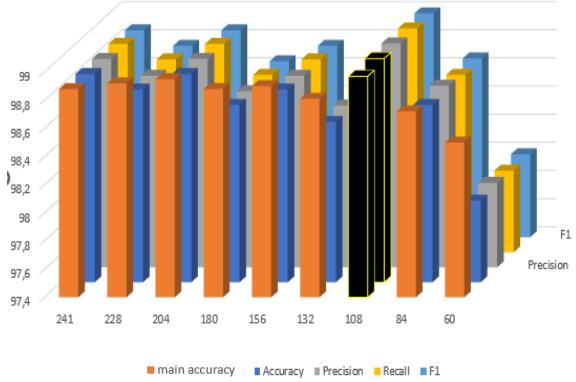
Discussion

- Graphe A4 : Meilleure Accuracy (99,43%) correspond à 204 attributs (85%), la meilleure Précision (99 ,66%) est obtenue avec 108 attributs(45%).
- Graphe A3 : Meilleure Accuracy de 98,77% avec 84 attributs(35%) puis se stabilise et Précision de 96,62% avec 84 attributs.
- Graphe A2 : avec une base de 180 attributs (75%) nous avons obtenu la Meilleure accuracy de 98,79% et la meilleure Précision de 98 ,73% .
- Graphe A1 : Meilleure Accuracy (96,62%) correspond à 156 attributs (65%), la meilleure Précision (97.3%) est obtenue avec 108 attributs(45%).

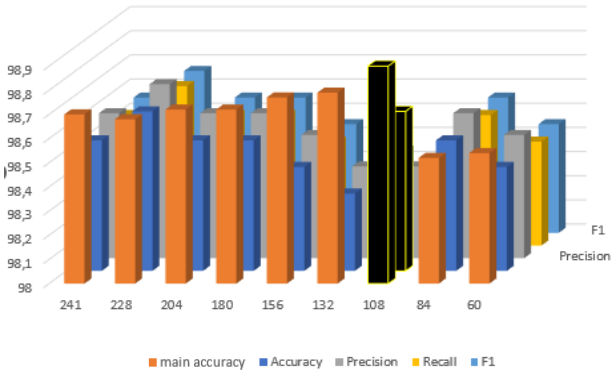
IV.1.2. Méthode 2 : mutuelle information



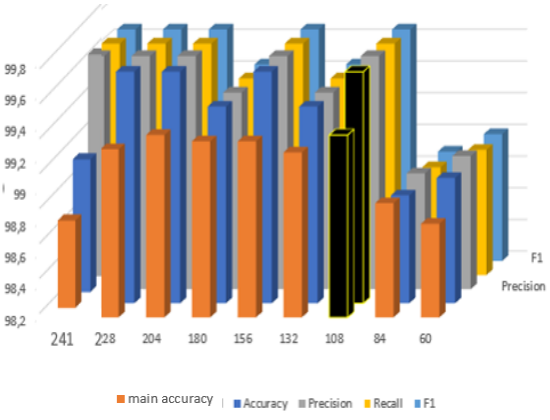
B1:k plus proches voisins



B1 :d'arbre de décision



B3 : Vecteur de Support



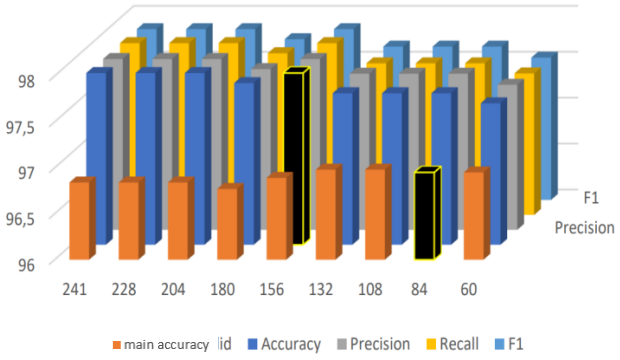
B4 : Forêt Aléatoire

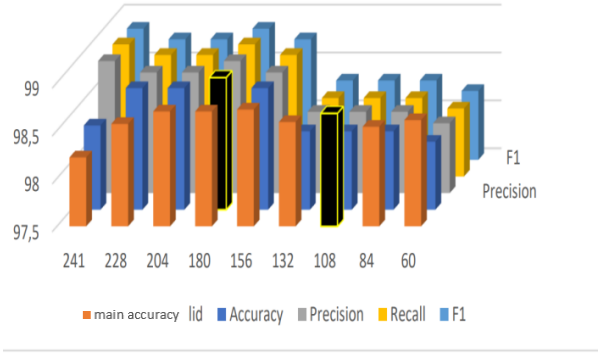
Figure 3.4 : Résultat de la méthode mutuelle information sur la base TUANDROMD

Discussion

- Graphe B4 : Meilleure Accuracy (99,36% avec 108 attributs) Précision de 99,5% avec 132 attributs
- Graphe B3 : Meilleure Accuracyde 98,87% avec 108 attributs Précision de 98,8% avec 108 attributs
- Graphe B2 : Meilleure Accuracyde 98,9% avec 108 attributs puis stable jusqu'à 132 attributs Précision de 98,8% avec 108 attributs
- Graphe B1 : Meilleure Accuracy la plus basse (96,5% avec 84 attributs) Précision de 96,7% avec 108 attributs

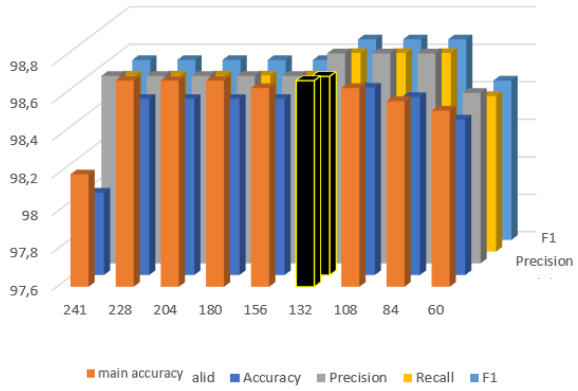
IV.1.3. Méthode 3 : Chi square

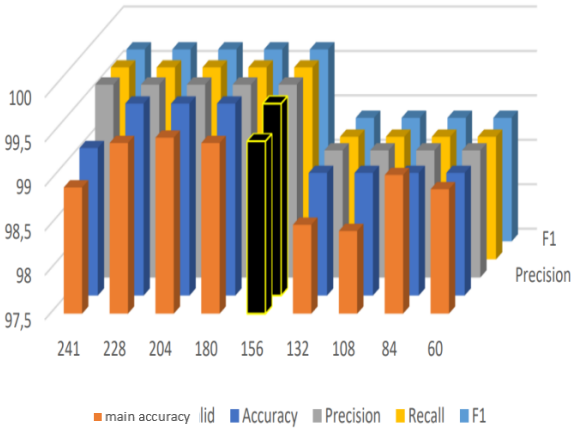




C1:k plus proches voisins

C2 :d'arbre de décision





C3 : Vecteur de Support

C4 : Forêt Aléatoire

Figure 3.5 : Résultat de la méthode Chi square sur la base TUANDROMD

Discussion

Graphe C4 :

- Avec une base de 84 attributs (35%) nous avons obtenu la Meilleure accuracy de (99,62%) et la meilleure Précision de 99,72% .

Graphe C3 :

- Avec une base de 132 attributs (55%) nous avons obtenu la Meilleure accuracy de 98,7% et la meilleure Précision de 98,69% .

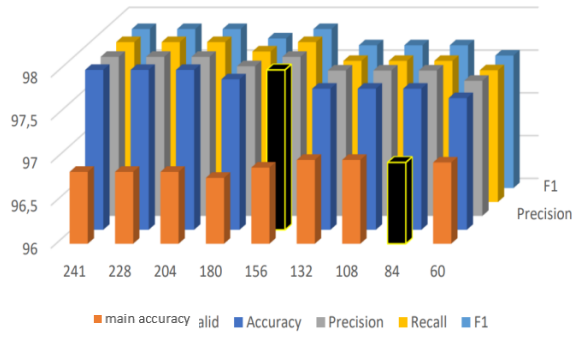
Graphe C2 :

- Meilleure Accuracyde 98,54% avec 108 attributs puis stable jusqu'à 132 attributs Précision de 98,84% avec 180 attribut

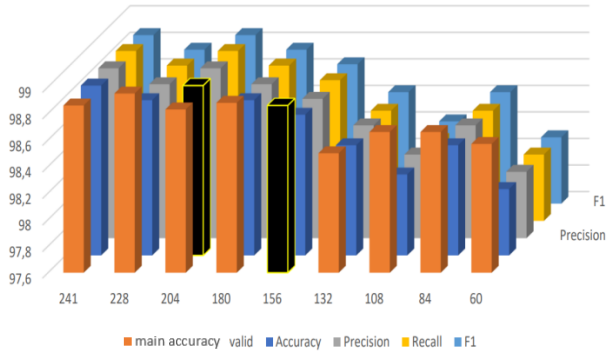
Graphe C1 :

- Meilleure Accuracy de 96,65% avec 84 attributs Précision de 97,45% avec 156 attributs

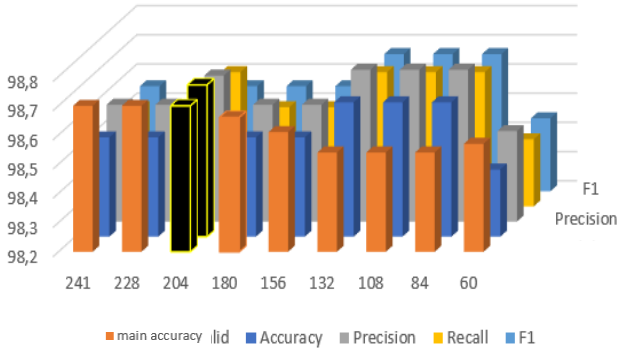
IV.1.4. Méthode 4 : Analyse de variance (ANOVA)



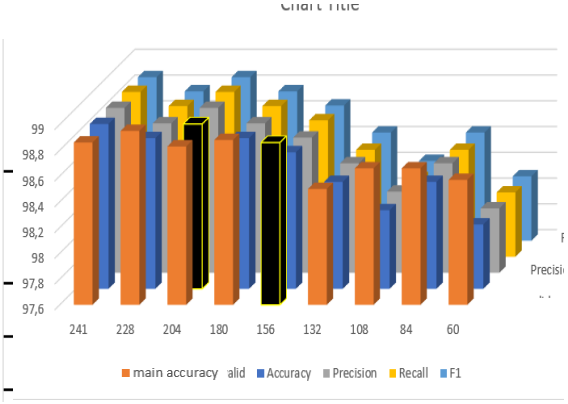
D1:k plus proches voisins



D2 :d'arbre de décision



D3 : Vecteur de Support



D4 : Forêt Aléatoire

Figure 3.6 : Résultat de la méthode ANOVA sur la base TUANDROMD

Discussion

Graphe D4 :

- Meilleure Accuracy de 98,88% avec 156 attributs Précision de 98,78% avec 204 attributs

Graphe D3 :

- Avec une base de 204 attributs (35%) nous avons obtenu la Meilleure accuracy de (98,7%) et la meilleure Précision de 98,67% .

Graphe D2 :

- Meilleure Accuracy de 97,54% avec 228 attributs Précision de 97,5% avec 204 attributs

Graphe D1 :

- Meilleure Accuracy de 98,30% avec 84 attributs Précision de 98,30% avec 156 attributs

Synthèse TUANDROM

Sur la base des informations fournies par les graphiques, on peut conclure que :

- Au fur et à mesure de la réduction du nombre d'attributs de 241 à 60 :
  1. La Accuracy augmente légèrement puis se stabilise
  2. Le temps d'entraînement diminue jusqu'à 25%
- Beaucoup d'attributs sont redondants ou non pertinents et peuvent être supprimés sans trop affecter la Accuracy mais en améliorant l'efficacité. Après 204 attributs, la Accuracy se stabilise jusqu'à 108 attributs, indiquant que les attributs supplémentaires n'améliorent pas significativement la Accuracy.
- La réduction du nombre d'attributs peut améliorer de manière significative l'efficacité des modèles sans affecter grandement leur Accuracy. Les méthodes RFE (RecursiveFeature Elimination) et MI (Mutuelle Information) , chi square , l'ANOVA montrent toutes que la Accuracy des modèles augmente légèrement au début lorsque le nombre d'attributs diminue, puis se stabilise ou diminue légèrement.

Dans l'ensemble, ces résultats démontrent l'importance de la sélection et de la réduction du nombre d'attributs pour améliorer l'efficacité des modèles d'apprentissage automatique.

IV.2. DATASET GENOME

Le tableaux suivant montre la taille correspond au différentes sélections effectuées :

Nombre d'attributs	4066	3862	3456	3049	2642	2236	1829	1423	1016
%	100	95	85	75	65	55	45	35	25
taille MB	24.8	23.56	21.8	18.6	16.12	13.64	11.16	8.68	6.2

Tableau 3.4: les différentes sélections effectuées sur GENOME

Après la sélection des attributs, les tailles du dataset peuvent être différentes en fonction des attributs conservés.

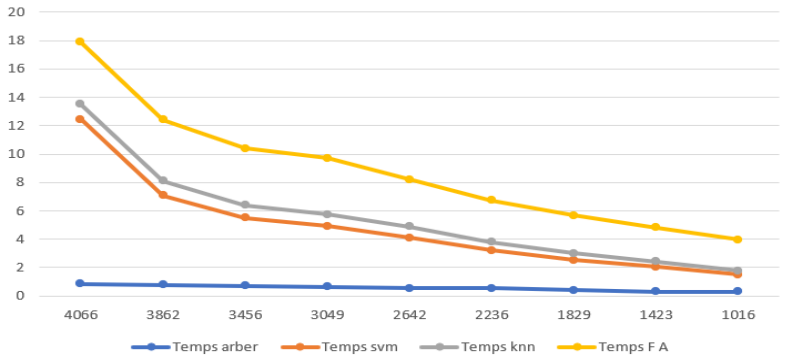
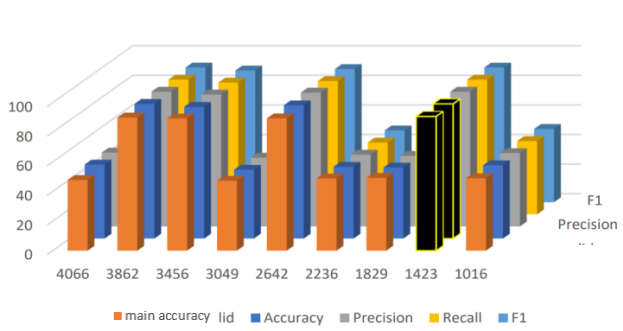
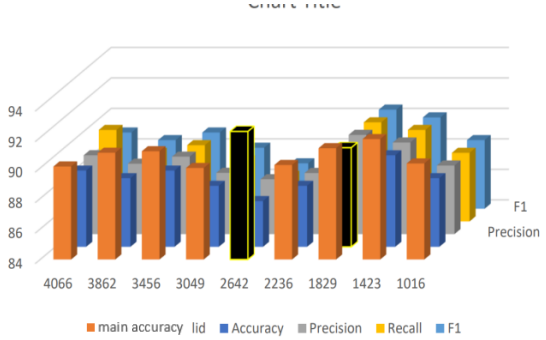


Figure 3.7 : temps exécution des algorithmes

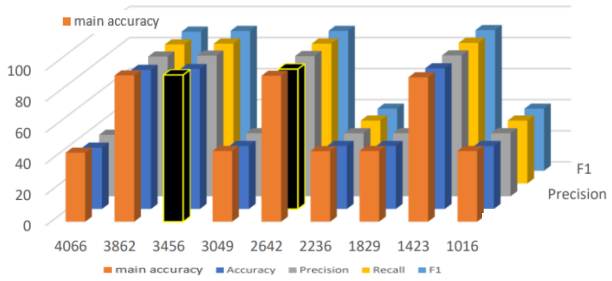
IV.2.4. Méthode 1 : Information mutuelle



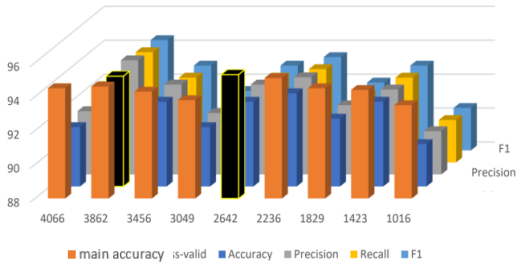
E1:k plus proches voisins



E2 :d'arbre de décision



E3 : Vecteur de Support



E4 : Forêt Aléatoire

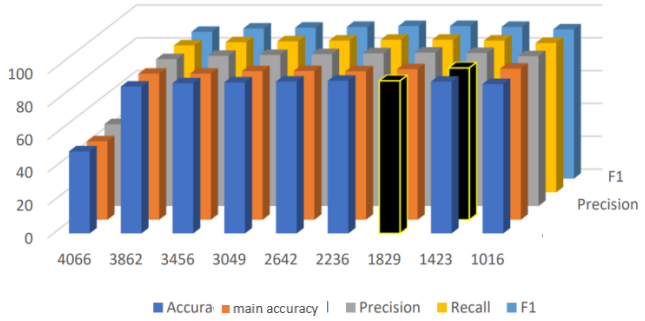
Figure 3.8 : Résultat de la méthode Information mutuelle sur la base Genome

Discussion

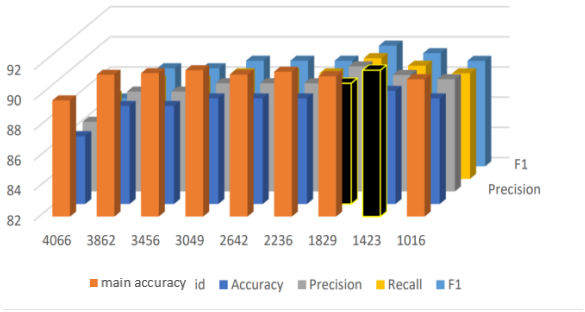
- Graphe E4 : Meilleure Accuracy (94%) correspond à 2642 attributs (65%), la meilleure Précision (93,2%) est obtenue avec 3863 attributs (95%)
- Graphe E3 : Meilleure Accuracy (84,2%) correspond à 3456 attributs (85%), la meilleure Précision (83,2%) est obtenue avec 2642 attributs (65%)

- Graphe E2 : Meilleure Accuracy (92,79%) correspond à 2642 attributs (65%), la meilleure Précision (92,59%) est obtenue avec 1829 attributs (45%)
- Graphe E1 : Avec une base de 1423 attributs (35%) nous avons obtenu la Meilleure accuracy de (84.8%) et la meilleure Précision de (84.2%).

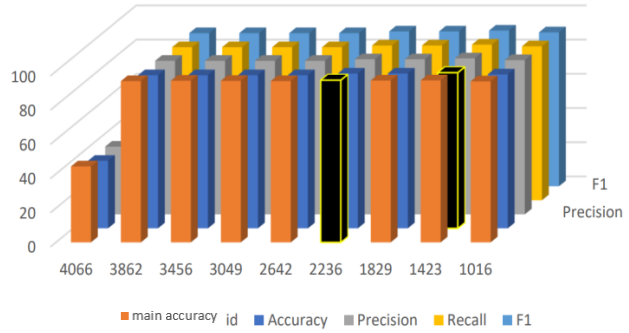
IV.2.2. Méthode 2 : Chi square



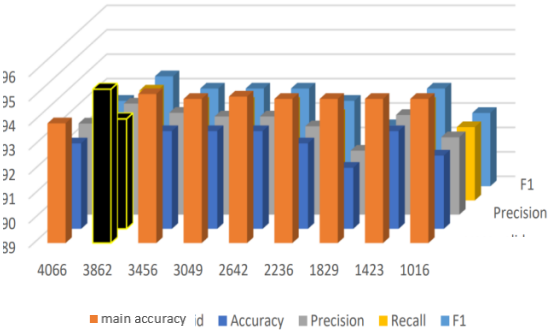
F1:k plus proches voisins



F2 :d'arbre de décision



F3 : Vecteur de Support



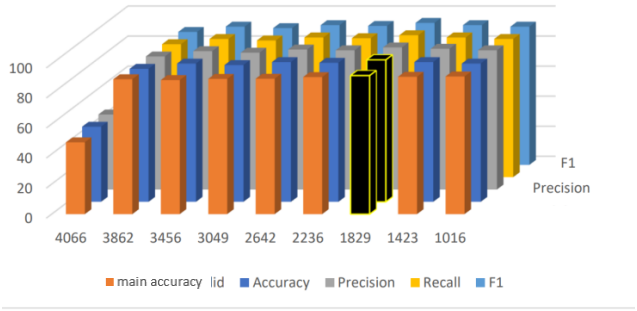
F4 : Forêt Aléatoire

Figure 3.9 : Résultat de la méthode Chi square sur la base Genome

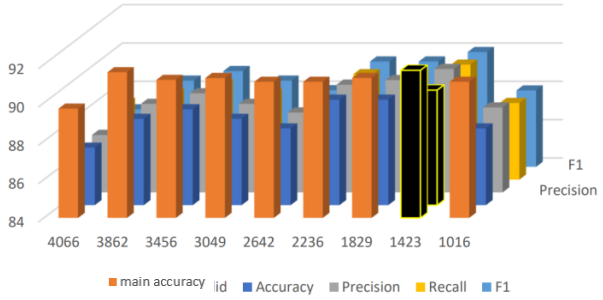
Discussion

- Graphe F4 : Avec une base de 3862 attributs (95%) nous avons obtenu la Meilleure accuracy de (93,18%) et la meilleure Précision de (93,43%).
- Graphe F3 : Meilleure Accuracy (81,7%) correspond à 2236 attributs (55%), la meilleure Précision (81,6%) est obtenue avec 1423 attributs (35%).
- Graphe F2 : Meilleure Accuracy (89,54%) correspond à 1423 attributs (35%), la meilleure Précision (89,14%) est obtenue avec 1423 attributs (35%).
- Graphe F1 : Meilleure Accuracy (79,65%) correspond à 1829 attributs (45%), la meilleure Précision (80,85%) est obtenue avec 1423 attributs (35%).

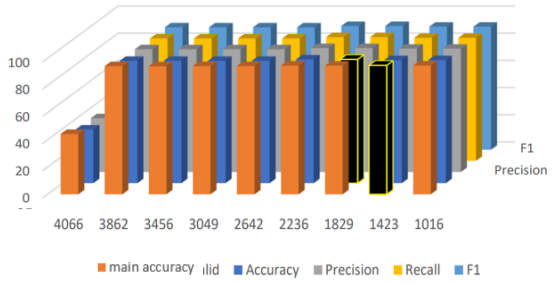
IV.2.3. Méthode 3 : Analyse de variance (ANOVA)



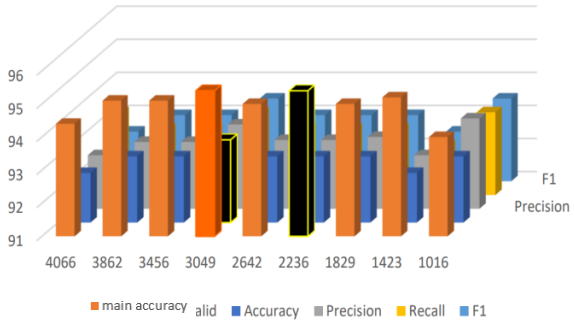
G1:k plus proches voisins



G2 :d'arbre de décision



G3 : Vecteur de Support



G4 : Forêt Aléatoir

Figure 3.10 : Résultat de la méthode ANOVA sur la base Genome

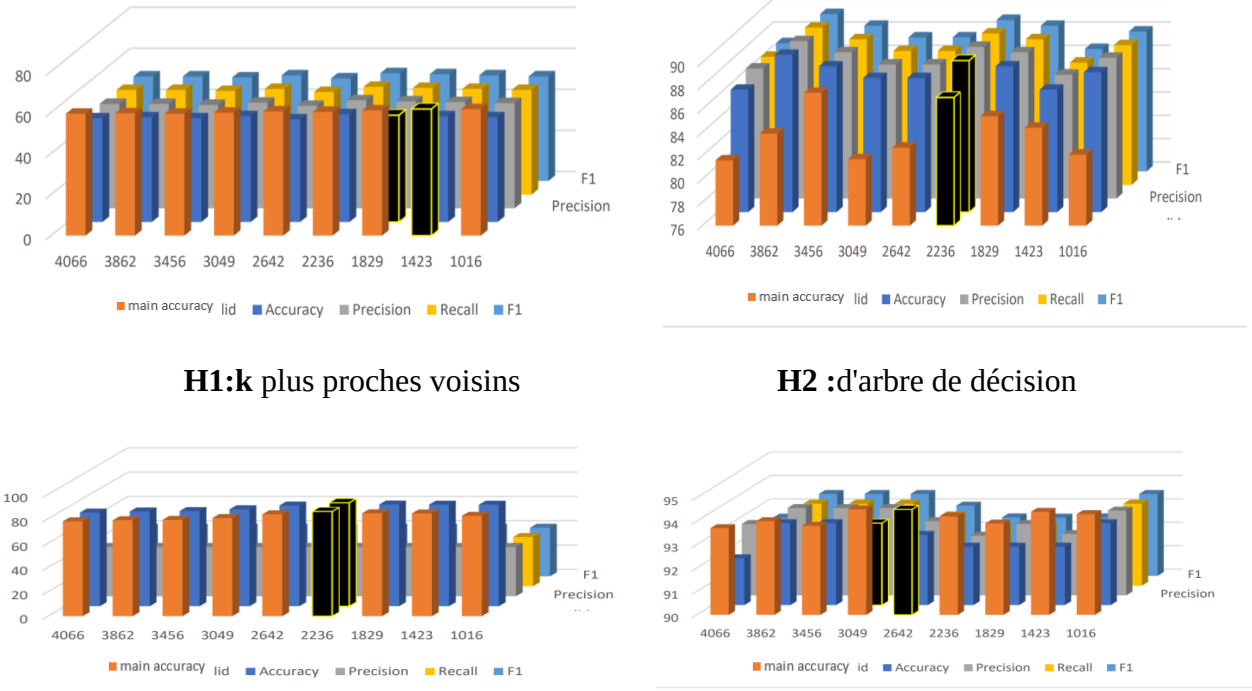
Discussion

- Graphe G4 : Meilleure Accuracy (93,1%) correspond à 2236 attributs (55%), la meilleure Précision (92,5%) est obtenue avec 3049 attributs (75%).
- Graphe G3 : Meilleure Accuracy (81,7%) correspond à 1423 attributs (35%), la meilleure Précision (81,71%) est obtenue avec 1829 attributs (45%).
- Graphe G2 : Avec une base de 1423 attributs (35%) nous avons obtenu la Meilleure accuracy de (89,7%) et la meilleure Précision de (89%).
- Graphe G1 : Avec une base de 1826 attributs (45%) nous avons obtenu la Meilleure accuracy de (79,70%) et la meilleure Précision de (79, 5%).

Au fur et à mesure de la réduction du nombre d'attributs de 4066 à 1016 :

- La Meilleure Accuracy augmente légèrement puis diminue

IV.2.4. Méthode 4 : Élimination récursive de caractéristiques (RFE)



H1:k plus proches voisins

H2 :d'arbre de décision

H3 : Vecteur de Support

H4 : Forêt Aléatoire

Figure 3.11 : Résultat de la méthode RFE sur la base Genome

Discussion

- Graphe H4 : Meilleure Accuracy (93,2%) correspond à 2642 attributs (65%), la meilleure Précision (92,3%) est obtenue avec 3049 attributs (75%).
- Graphe H3 : Avec une base de 2236 attributs (55%) nous avons obtenu la Meilleure accuracy de (82,87%) et la meilleure Précision de (82,77%).
- Graphe H2 : Avec une base de 2236 attributs (55%) nous avons obtenu la Meilleure accuracy de (88,9%) et la meilleure Précision de (87,9%).
- Graphe H1 : Meilleure Accuracy (69,5%) correspond à 2236 attributs (55%), la meilleure Précision (68,5%) est obtenue avec 1829 attributs (45%).

Synthèse TUANDROM

Au fur et à mesure de la réduction du nombre d'attributs de 4066 à 1016 :

- La Accuracy augmente légèrement puis se stabilise
- Le temps d'entraînement diminue jusqu'à 25%
- Le temps d'entraînement diminue jusqu'à 10%

Beaucoup d'attributs sont redondants ou non pertinents et peuvent être supprimés sans trop affecter la Accuracy mais en améliorant l'efficacité. Après 3456 attributs, la Accuracy se stabilise jusqu'à 1423 attributs, indiquant que les attributs supplémentaires n'améliorent pas significativement la Accuracy et Précision.

IV.3. Méthode proposée :

Cette étude expérimentale a montré que tous les bons résultats ont été obtenu avec un nombre d'attribut inférieurs a168 attributs Dans la base TUANDROME et nombre d'attribut inférieurs a 3252 attributs dans la base GENOME dataset

L'étape suivant de ce travail et de faire une combinaison de deux méthodes de sélection (mutule – information, chi square).

La première méthode de sélection sera applique avec un pourcentage de 80% des attributs (3252)**,** ensuit nous appliquant plusieurs sélections sur cet ensemble en utilisant chi square.

Les méthodes de sélection de l'information mutuelle et du chi-square ont été choisies car elles ont données les meilleurs résultats dans cette étude expérimentale.

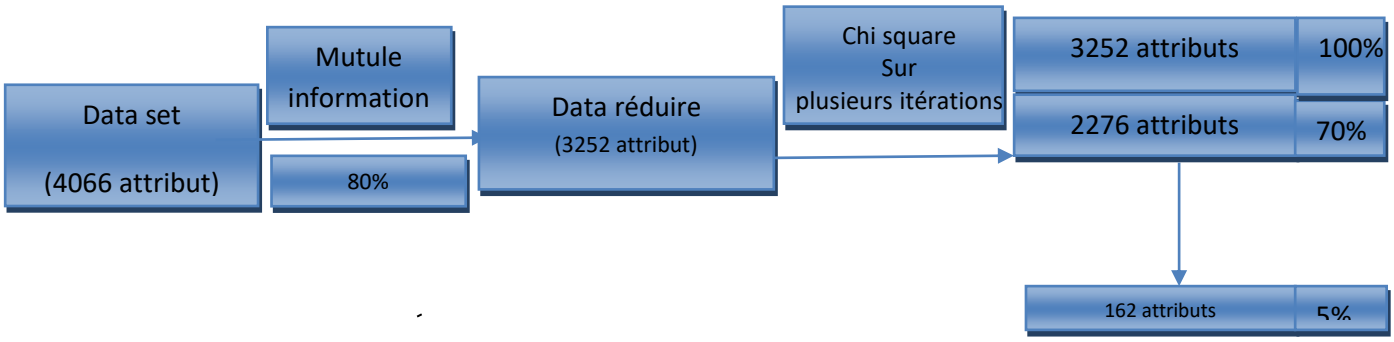


FIGURE 3.12 : structure générale de Méthode proposée

Le processus proposé passe par deux étapes fondamentales

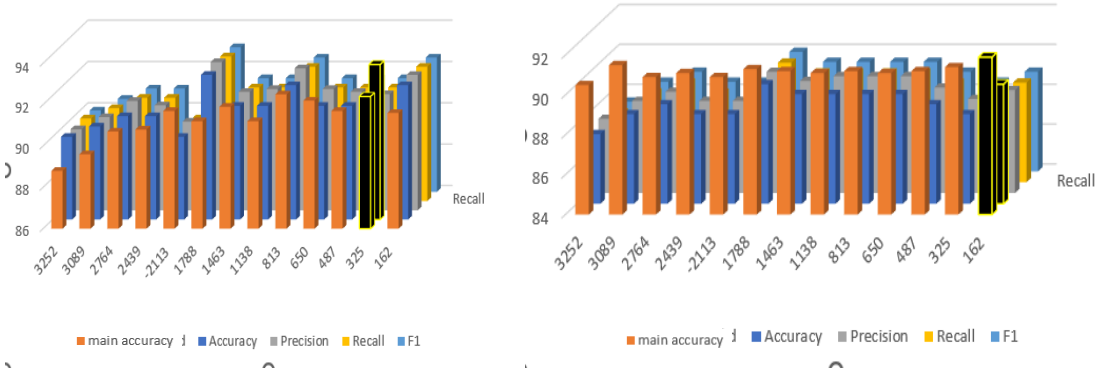
Étape 1 (Information mutuelle) :

- Appliquer l'information mutuelle pour sélectionner les attributs pertinents
- Sélectionner les 3252 premiers attributs (80% de tous les attributs) à partir de l'ensemble de données complet (4066 attributs).

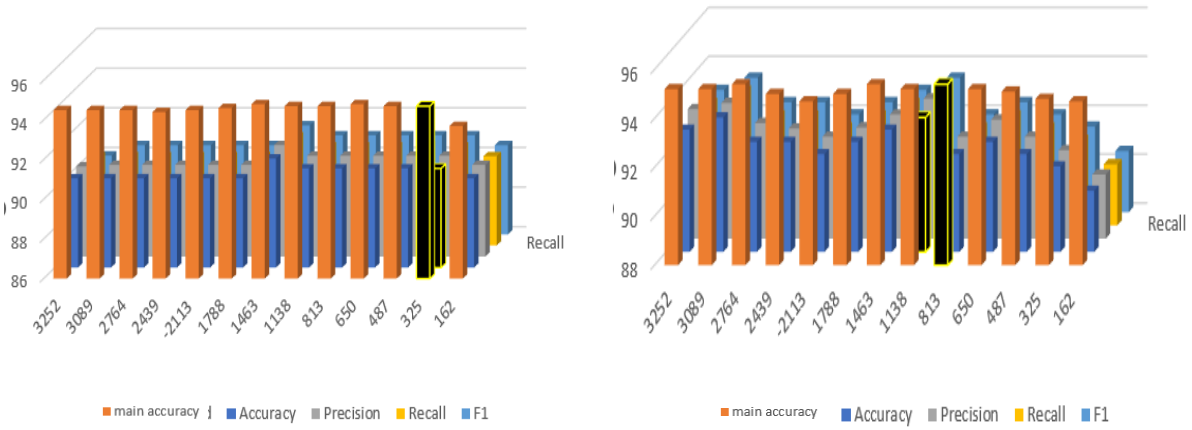
Étape 2 (Test du chi-square) :

- Appliquer le test du Chi-square pour sélectionner davantage les attributs importants parmi les 3252
- Sélectionner les maximum attributs ayant les scores de Chi-square les plus élevés.

IV.3.1 .Résultats obtenus**:**



N1:k plus proches voisins **N2 :**d'arbre de décision



N3 : Vecteur de Support **N4 :** Forêt Aléatoire

Figure 3.13 : Résultat de la Méthode proposée sur la base GENOME

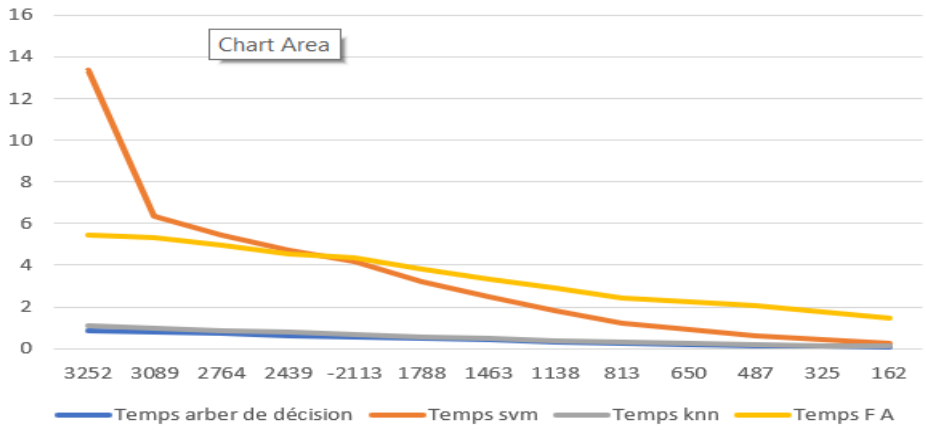


Figure 3.14 : temps des algorithmes

la figure ci-dessus montre que le temps d'exécution se diminue avec la réduction du nombre d'attributs

IV.3.2 Évaluation de méthode proposée

	Mutule information			Chi square			Méthode proposée		
Les algorithmes	**Accurcy**	**précision**	**number attributs**	**accurcy**	**Precision**	**number attributs**	**Accurcy**	**précision**	**numbe attributs**
Kpp-v	84,8%	84,2%	1423	79,65%	80,85%	1423	93,42%	93,76%	325
Arbre de décision	92,79%	92,59%	2642	89,54%	89,14%	1423	92,95%	92,9%	162
SVM	84,2%	83,2%	3456	81,7%	81,6%	2236	91,59%	92,67%	325
Forêt Aléatoire	94	93,2%	3863	93,43%	93,28%	3863	94%	93,5%	813

Tableau 3.3: Évaluation de méthode proposée

Nous observons une amélioration remarquable des performances de la méthode proposée :

- **KPP-V :**
  1. Réduction de la dimension de la base sélection de 325 parmi 4066 (8%)
  2. La précision obtenue est égale à 93,76% et L'accuracy est 93,42% par contre en utilisant une seul méthode l' accurcy maximale obtenue est 84,8% , la précision maximal est 84,2% et un nombre d'attribut important (1423) .
- **Arbre de décision :**
  1. Réduction de la dimension de la base sélection de 162 parmi 4066 (4%)
  2. L' accuracy 92,95% et précision 92.9% par contre en utilisant une seul méthode nous avons obtenu : une accurcy maximale de 92,79% , la précision maximale 92,59% et nombre d'attribut important (2642) .

- **SVM :**

1. Réduction de la dimension de la base sélection de 325 parmi 4066 (8%)
2. Les résultats obtenus : Accuracy 91,59 et un taux de 92,67 de précision . En utilisant une seul méthode nous avons obtenu : la précision maximale est 83,2% , taux de précision de 84,2% et un nombre dattribut important (3456) .

- **Forêt Aléatoire :**

1. Réduction de la dimension de la base sélection de 813 parmi 4066 (20%)
2. Les résultats obtenus : Accuracy 94% et un taux de 93,5 de précision . En utilisant une seul méthode nous avons obtenu : la précision maximale est 93,43% , taux de précision de 93,28% et un nombre dattribut important (3863) .

IV.3.3 Présentation de l'application :

Notre application intitulé "Data Analyzer" offre diverses fonctionnalités pour l'analyse de données et la génération de rapports. Elle propose une interface conviviale avec deux onglets principaux : "Génération de rapports" et "Générateur CSV".

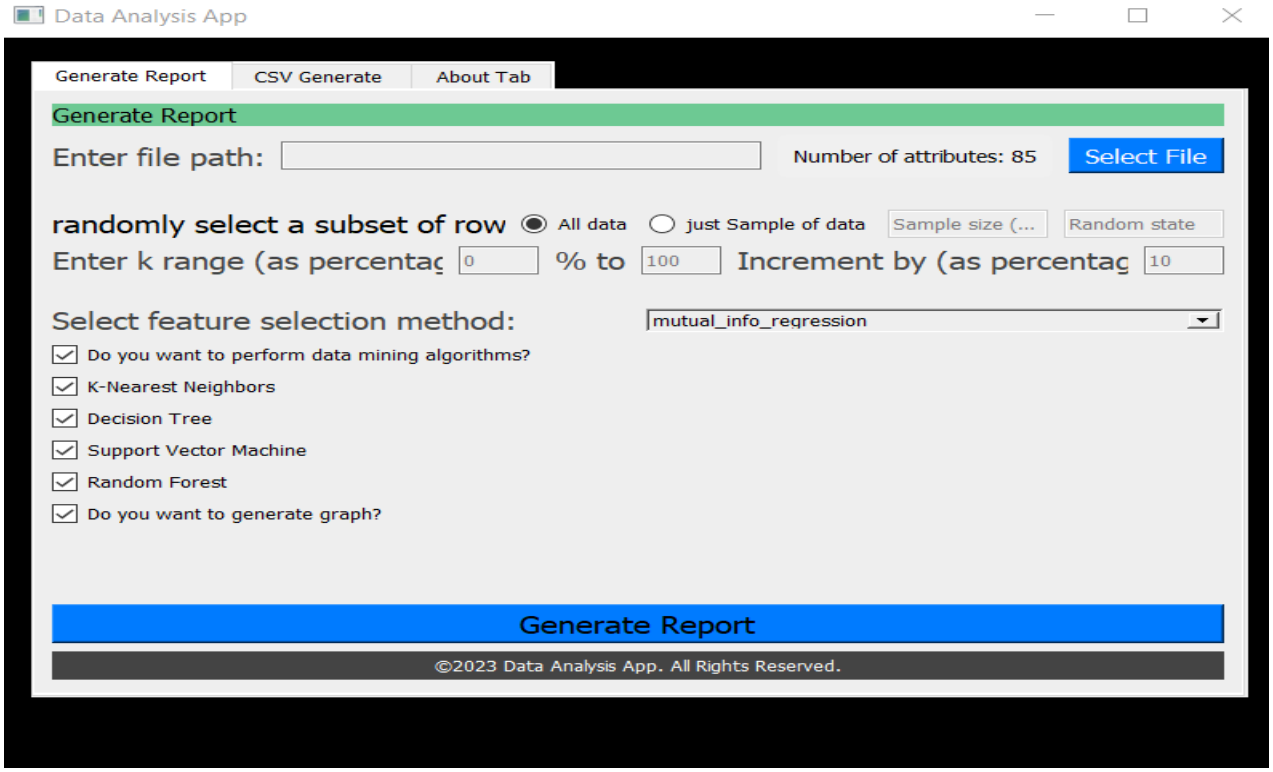


Figure 3.15 : interface appliction Generate Raport tab

I) Onglet Génération de rapports :

Cet onglet permet aux utilisateurs de générer des rapports basés sur leurs données. Pour utiliser cette fonctionnalité, les utilisateurs doivent spécifier le chemin du fichier contenant les données à analyser. Ils peuvent choisir entre deux ensembles de données : "Données d'échantillon" ou "Toutes les données".

L'application propose des méthodes de sélection d'attributs pour permettre aux utilisateurs de choisir les variables à inclure dans l'analyse. Les méthodes de sélection d'attributs disponibles sont les suivantes :

1. Classification de l'information mutuelle
2. Test du chi-square
3. Analyse de la variance (ANOVA) - valeur F
4. Régression - valeur F
5. Régression de l'information mutuelle

De plus, les utilisateurs peuvent définir une plage de valeurs pour la variable "k" et sélectionner les algorithmes d'extraction de données à utiliser. Ils ont également la possibilité de générer des graphiques basés sur les données analysées. Une fois tous les paramètres configurés, les utilisateurs peuvent lancer le processus1829 d'analyse et de génération de rapports en cliquant sur le bouton "Générer le rapport".

Si l'option "Données d'échantillon" est sélectionnée, l'application choisira aléatoirement un sous-ensemble de lignes du jeu de données en fonction de la taille d'échantillon spécifiée. Les résultats de l'analyse seront ensuite affichés sous forme de rapport, et les utilisateurs pourront également visualiser les résultats à l'aide de graphiques.

II) Onglet Générateur CSV

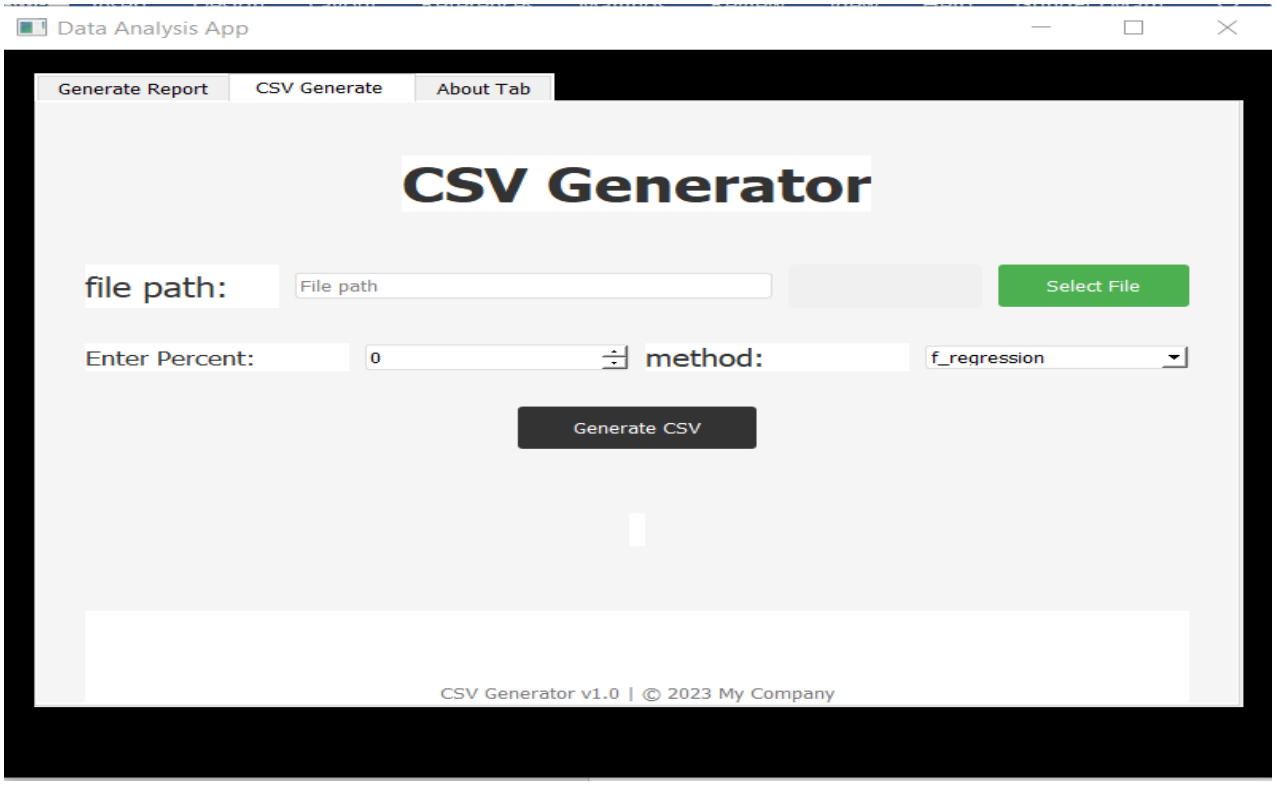


Figure 3.16 : interface appliction CSV Generator tap

L'onglet Générateur CSV permet aux utilisateurs de générer un nouveau fichier CSV ne contenant que les fonctionnalités sélectionnées en fonction de l'attribut et de la méthode choisis. Cette fonctionnalité est particulièrement utile lors de l'analyse de grands ensembles de données où toutes les fonctionnalités ne sont pas pertinentes ou informatives.

Les méthodes de sélection d'attributs disponibles dans l'onglet Générateur CSV sont les mêmes que celles mentionnées ci-dessus :

1. Classification de l'information mutuelle
2. Test du chi-square
3. Analyse de la variance (ANOVA) - valeur F
4. Régression - valeur F
5. Régression de l'information mutuelle

Les utilisateurs peuvent sélectionner l'attribut et la méthode souhaités, puis lancer le processus de réduction du fichier CSV en cliquant sur le bouton correspondant. Cela générera un nouveau fichier CSV ne contenant que les fonctionnalités sélectionnées, ce qui permettra aux utilisateurs d'effectuer une analyse plus ciblée et efficace sur leurs données. En conclusion, l'application "Data Analyzer" propose un ensemble complet d'outils pour l'analyse de données et la génération de rapports. L'onglet Génération de rapports permet aux utilisateurs d'analyser des données, de générer des rapports et de visualiser les résultats, tandis
que l'onglet Générateur CSV offre une solution pratique pour réduire la taille des ensembles de données tout en conservant les fonctionnalités pertinentes

Conclusion :

Ce chapitre a démontré l'efficacité de la sélection des caractéristiques pour améliorer les performances des modèles d'apprentissage automatique. Les résultats ont montré que la nouvelle méthode de sélection en deux étapes utilisant l'information mutuelle et le test du chi-square pouvait identifier un nombre optimal d'attributs pertinents pour atteindre un niveau de précision élevé et un temps d’exécution réduit.

Conclusion générale :

Dans cette étude, nous avons examiné le problème de la sélection des attributs dans le contexte de la détection des applications Android malveillantes. Nous avons comparé quatre méthodes populaires de sélection des attributs, à savoir l'information mutuelle, le chi-square (χ^2), l'analyse de variance (ANOVA) le l'élimination récursive des caractéristiques, afin d'identifier la technique la plus efficace pour sélectionner les attributs dans le cadre de la détection des applications Android malveillantes.

Sur la base de nos résultats expérimentaux et de notre analyse, nous formulons les conclusions suivantes :

• L'information mutuelle a démontré des performances supérieures dans la sélection des attributs pour la détection des applications Android malveillantes. Elle a systématiquement surpassé les autres méthodes, ce qui indique son efficacité pour identifier les caractéristiques les plus pertinentes.

• Le chi-square (χ^2) et l'ANOVA ont également donné des résultats prometteurs, montrant leur potentiel pour la sélection des attributs dans la détection des malwares Android. Bien qu'ils ne soient pas aussi efficaces que l'information mutuelle, ils ont néanmoins fourni des informations précieuses sur les attributs pertinents.

• l'élimination récursive des caractéristiques a également montré des résultats intéressants, bien que moins performants que les autres méthodes. Cependant, cette approche présente l'avantage de fournir un processus itératif permettant de réduire le nombre d'attributs sans dépendre d'une mesure spécifique de pertinence.

Les implications de nos résultats sont significatives pour le développement de systèmes de détection de logiciels malveillants robustes. En utilisant des méthodes efficaces de sélection des attributs, comme l'information mutuelle, nous pouvons améliorer la précision et l'efficacité de l'identification des applications Android malveillantes, ce qui contribue à renforcer la confidentialité et la sécurité des utilisateurs.

En combinant l'information mutuelle et les tests du chi-square, nous avons observé une amélioration des performances par rapport aux méthodes individuelles. La sélection du nombre optimal d'attributs en fonction du score de précision principale a permis de maximiser la précision tout en réduisant le nombre d'attributs, ce qui améliore l'interprétabilité du modèle. En se concentrant uniquement sur le score de précision principale. La nouvelle méthode présentée dans cette étude offre des implications prometteuses pour la sélection des attributs dans le contexte de la détection des applications Android malveillantes. Son approche en deux étapes améliore à la fois les performances et l'interprétabilité des classificateurs en identifiant un ensemble optimal d'attributs pertinents.

Comme perspective nous envisageons :

- d'appliquer l'hybridation proposée sur d'autre jeux de données .
- d'étudier les méthodes d'extraction d'attributs et de proposer une nouvelle méthode pour minimiser le nombre d'attributs.
- de tester d'autre hybridation pour minimiser au maximum la taille de la base.

Bibliographies:

[1] A. L. Blum, P. Langley, "Selection of relevant features and examples in machine learning," *Artif.Intell*, vol. 97, no. 1, pp. 245-271, 1997.

[2] Réda Mohamed HAMOU. « Réduction de dimension et similarité ».In: *Polycopié*

.**[3]** Yang, Y., and Pedersen, J.O. (1997). A comparative study on feature selection in text categorization. In Icml,

[4] Fer chichis(2008) «sélection de variables et de caractéristique pour une méthode d'apprentissage Masters thesis, Ecole Nationale d'Ingénieurs de Tunis»

[5] R. Gutierrez-Osuna, "Introduction to Pattern Analysis," Texas A&M University, lecture 11 2012.

[6] H. Liu, L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *Knowl. Data Eng. IEEE Trans. On*, vol. 17, no. 4, pp. 491–502, 2005.

[7] G. Wang, Q. Song, H. Sun, X. Zhang, B. Xu, Y. Zhou, "A Feature Subset Selection Algorithm Automatic Recommendation Method," *Journal of Artificial Intelligence Research*, vol. 47, 2013.

[8] M. Dash et H. Liu. Feature selection for classification. Intelligent Data Analysis, pages 131–156, 1997.

[9] Rainie, H., Corn_eld, M., & Horrigan, J. B. (2005). The Internet and campaign 2004. Pew Internet & American Life Project...

[10] R. Kohavi, G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, no. 1, pp. 273–324, 1997.

[11] Kohavi, R., and John, G.H. (1997). Wrappers for feature subset selection. Artif. Intel.97, 273{324.

[12] S. El Ferchichi, "Sélection et Extraction d'attributs pour les problèmes de classification, «Université Lille1 des Sciences et Technologies Université de Tunis El Manar Ecole Nationale d'Ingénieurs de Tunis, phD thèses 2013..

[13] Y. Saeys, I. Inza, P. Larrañaga, "A review of feature selection techniques in bioinformatics," *bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.

[14] S. F. Pratama, A. K. Muda, Y.-H. Choo, N. A. Muda, "A Comparative Study of Feature Selection Methods for Authorship Invarianceness in Writer Identification," *Int. J. Comput. Inf. Syst. Ind. Manag. Appl.*, vol. 4, pp. 467–476, 2012.

[15] H. Almuallim ET T.G. Dietterich: Learning with many irrelevant features. In Proceedings of the Ninth National Conference on Artificial Intelligence, pages 547–552, 1991.

[16] S.Nakariyakul, "A Review of Suboptimal Branch and Bound Algorithms," *IPCSIT*, vol. 2, 2011.

[17] H. Chouaib, "Sélection de caractéristiques: méthodes et applications," Université Paris Descartes, 2011.

[18] M. A. Ibrahim, O. A. Ojo, and P. A. Oluwafisoye, "On feature selection methods for accurate classification and analysis of emphysema ct images," *Int. J. Med. Imaging*, vol. 5, p. 70, 2018.

[**19]** A. W. Whitney, "A direct method of nonparametric measurement selection," *IEEE Transactions on Computers*, vol. 100, pp. 1100-1103, 1971..

[20] R. Setiono, "Feature selection and classification-a probabilistic wrapper approach," in *proceedings of the 9 International Conferences on Industrial and Engineering Applications of Al and ES*, 1996.

.**[21]** H. Liu and R. Setiono, "Incremental feature selection," *Applied Intelligence*, vol. 9, pp. 217-230, 1998.

[22] A. K. Mohanty, M. R. Senapati, and S. K. Lenka, "A novel image mining technique for classification of mammograms using hybrid feature selection," *Neural Computing and Applications*, vol. 22, pp. 1151-1161, 2013.

[23] I. Ned jar, M. EL HABIB DAHO, N. Settouti, S. Mahmoudi, and M. A. Chikh, "RANDOM FOREST BASED CLASSIFICATION OF MEDICAL X-RAY IMAGES USING A GENETIC ALGORITHM FOR FEATURE SELECTION," *Journal of Mechanics in Medicine and Biology*, vol. 15, p. 1540025, 2015.

[24] Jensen R., “Combining rough and fuzzy sets for feature selection”, these de Doctorat, University de Edinburgh, Royaume-Uni, 2005

[25] Aghdam M. H., Ghasem-Aghaee N., Basiri M. E., “Text feature selection Using ant colony optimization”, Expert Systems with Applications Vol. 36, Issue 3, Part 2, pp 6843–6853, 2009.

[26] Meena M.J., Chandran K.R., Karthik A., Samuel A.V., "An enhanced ACO Algorithm to select features for text categorization and its parallelization", Expert Systems with Applications, Vol. 39, pp 5861–5871. 2012.

[27] Grandidier F., "Un nouvel algorithme de sélection de caractéristiques-Application à la lecture automatique de l'écriture manuscrite", Thèse de Doctorat, Université du Québec, Canada 2003.

[28] Saeys Y., Inza I., Larranaga P., "A review of feature selection techniques in bioinformatics", Bioinformatics, Vol. 23, issue 19, pp 2507–2517, 2007.

[29] https://www.vmware.com/fr/topics/glossary/content/application-security.html

[30] La documentation pour les développeurs Android fournit des informations détaillées sur les caractéristiques de sécurité, Site web : https**://developer.android.com/guide/topics/security/**

[31] John Doe, "Understanding Android App Permissions: A Comprehensive Study", Source: Proceedings of the International Conference on Mobile Applications (ICMA), 20XX

[32] Les permissions sous Android : https://blog.rolandl.fr/2016-02-13-les-permissions-sous-android-1-slash-6-android-et-les-permissions.html

[33] Android Développer (2017). System Permissions.

Repéré à http://developer.android.com/guide/topics/security/permissions.html

[34] Benjamin Morin. Modèle de sécurité d'Android. MISC 51, 2010.

[35] Jon Oberheide http://jon.oberheide.org/blog/2010/06/25/remote-kill-and-install-on-google-android/

[36] Jon Oberheide http://jon.oberheide.org/blog/2010/06/28/a-peek-inside-the-gtalkservice-connection/

[37] "Mobile Application Security Threats and Countermeasures: A Survey" - Research Gate (2018) Disponible ici: https://www.researchgate.net/publication/329918606_Mobile_Application_Security_Threats_and_Countermeasures_A_Survey

[38] Mohit Singhal. Analyse et catégorisation des logiciels malveillants de téléchargement Drive-by _a l'aide de Sandboxing et de l'ensemble de règles Yara. PhD thesis, Texas, 2019.

[39] Kateryna Chumachenko.Machine learning methods for malware detection and classification.phD thesis, Kaakkois-Suomen ammattikorkeakoulu, 2017.

[40] Philippe Beau camps. Analyse de programmes malveillants par abstraction de com.-portements. PhD thesis, France, INPL, 2011.

[41] Zahra Bazrafshan, Hashem Hashemi, Seyed Mehdi Hazrati Fard, and Ali Hamzeh. A survey on heuristic malware detection techniques. The 5th Conference on Information and Knowledge Technology, IEEE, 113{120, 2013.

[42] Annie H Toderici and Mark Stamp. Chi-squared distance and metamorphic virus

Detection. Journal of Computer Virology and Hacking Techniques-Springer-,

9(1):1{14, 2013}.

[43] Ed Skoudis and Lenny Zeltser. Malware: Fighting malicious code. Prentice Hall

Professional, 2004, pages 164.

[44] Mohamed BELAOUED. Approches Collectives et Coopératives en Sécurité des Systèmes Informatiques. PhD thesis, Skikda, 2016.

[45] Grégoire Jacob, Herve Debar, and Eric Filiol. Behavioral detection of malware: from a survey towards an established taxonomy. Journal in computer Virology-Springer-, 4(3) :{251,266}, 2008.

[46] Rieck,k,Holz,Willems,c,Dussel,p,& Laskov,p.(2008).Learning and classification of malware behaviour. In International Conference on Knowledge Discovery and Data mining (pp.1088-1096).springe

[47] Mohamad Baset. MACHINE LEARNING FOR MALWARE DETECTION

.Heriot-Watt University. Page 18, 2016.

[48] Mustafa A.Ali, wesam S.Bhaya. Review on Malware and Malware Detection Using Data Mining Techniques. Journal of University of Babylon for Pure and Applied Sciences, November 2017.

[49] Matthew G. Schultz and Eleazar Eskin, Erez Zadok, Salvatore J. Stolfo. *Data Mi-ning Methods for Detection of New Malicious Executables*. IEEE Xplore, Conference Paper 10.1109/SECPRI.2001.924286, 2001.

[50] Y.Zhenlong. Droid-sec: deep learning in android malware detection. SIGCOMM '14 Proceedings of the ACM conference on SIGCOMM Chicago, Illinois, USA, 2014.

[51] S.Justin, K. (2012). *A machine learning approach to android malware detection*.

European Intelligence and Security Informatics Conference IEEE.

[52] S.Asaf, K. (2012). *A behavioural malware detection framework for android*

devices. Journal of Intelligent Information Systems Springer.

[53] P.Naser, Z. (2013). *Machine learning for android malware detection using*

Permission and api calls. IEEE 25th International Conference on Tools with

Artificial Intelligence.

[54] B.Shaikh, M. (2015). *A novel approach to detect android malware*. Procedia

Computer Science 45, Peer review under responsibility of scientific committee

Of International Conference on Advanced Computing Technologies and Applications

(ICACTA-2015), Elsevier B.V.

[55] C.Li, K. (2016). *Android malware detection based on factorization machine*.

Cryptographie and Security (cs.CR

[56] H.Fereidooni, M. (2016). *Android malware detection using static analysis of*

Applications. IFIP International Conference on New Technologies, Mobility and

Security (NTMS) IEEE.

[57]. archive. [Online] [Cited: 02 1, 2023.] https://archive.ics.uci.edu/ml/datasets/TUANDROMD+%28+Tezpur+University+Android+Malware+Dataset%29#.

[58] Dissecting Android Malware: Characterization and Evolution. [Online] [Cited: 04 2, 2023.] http://www.malgenomeproject.org/.

[59] Information mutuelle - Définition. *techno-science.* [Online] 5 03, 2023. https://www.techno-science.net/definition/6367.html.

[60] Le test de Chi square (X2) . classesbranchees.csf.bc.ca. [Online] [Cited: 0,3 23, 2023.] https://classesbranchees.csf.bc.ca/bi-jv/wp-content/uploads/sites/15/1.-Le-test-de-Chi-Carre%CC%81.pdf.

[61] ANOVA - Analyse de variance à un et à deux facteurs. cours-F. [Online] [Cited: 02 21, 2023.] https://webdemo.inue.uni-stuttgart.de/webdemos/02_lectures/nachrichtentechnik/f-tests-anova

[62] pandas-python-data-science. pandas-python-data-science. [Online] [Cited: 02 21, 2023.] https://datascientest.com/pandas-python-data-science