REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE 20 AOUT 1955 – SKIKDA

Faculté des Sciences

Département d'Informatique

Mémoire de fin d'études en vue de l'obtention du diplôme de Master en Informatique

<u>Option</u> : Réseaux et Systèmes Distribués (RSD)

Thème

Détection de Spam par les Techniques de l'Intelligence Artificielle (IA)

Réalisé par :
BALASKA Zahia
KHALDI Sabrina

Encadré par :
Dr. NAFIR Abdenacer
Dr. MAZOUZI Smaine

Session : Juillet 2024

Dédicace

Tout d'abord, je tiens à remercier DIEU

De m'avoir donné la force et le courage de mener

À bien ce modeste travail.

Je tiens à dédier cet humble travail à :

À mes chers parents qui n'ont jamais cessé de m'aimer et m'encourager durant tout mon

parcours et qui m'ont permis d'être aujourd'hui ce que je suis.

À Toi ma chère mère

À Toi mon cher Père

À mes sœurs et mon frère

À toute la famille.

A ma camarade: KHALDI Sabrina

AUX encadrantS Dr.NAFIR Abdenacer et Dr.MAZOUZI Smaine

A tous mes collègues de deuxième année master.

MERCI A TOUS.

Melle. BALASKA Zahia

Dédicace

Tout d'abord, je tiens à remercier DIEU

De m'avoir donné la force et le courage de mener

À bien ce modeste travail.

Je tiens à dédier cet humble travail à :

À mes chers parents qui n'ont jamais cessé de m'aimer et m'encourager durant tout mon

parcours et qui m'ont permis d'être aujourd'hui ce que je suis.

À Toi ma chère mère

À Toi mon cher Père

À mes sœurs et mon frère

À toute la famille.

A ma camarade: BALASKA Zahia

ÀUX encadrantS Dr.NAFIR Abdenacer et Dr.MAZOUZI Smaine

A tous mes collègues de deuxième année master.

MERCI A TOUS.

Melle. KHALDI Sabrina

Remerciement

Nous tenons à remercier avant tout le bon Dieu, Le Tout Puissant, de nous avoir donné la foi, la volonté et le courage de mener à bien ce modeste travail.

Nous adressons le grand remerciement à nos encadreurs, Docteur NAFIR Abdenacer et Docteur MAZOUZI Smaine, pour nous avoir toujours faits confiance quant à la façon de mener nos travaux.

Nous tenons tout d'abord à leurs témoigner notre reconnaissance pour leurs suivi rigoureux et leurs grande disponibilité. Nous leurs présentons également nos remercîments pour leurs aide, leurs soutien, leurs judicieux conseils et orientations.

Merci à nos mères & péres qui nous sont les plus chères.

Finalement, nos vifs remerciements vont à l'ensemble de nos enseignants et à toute personne ayant contribué de près ou de loin à l'élaboration de ce travail

Aux membres du jury pour avoir accepté de bien vouloir lire notre travail, l'examiner, l'évaluer et nous corriger à mener à bien la réalisation de ce modeste travail.

Résumé

Au cours des dernières décennies, l'utilisation du courrier électronique s'est généralisée, entraînant des spams ou des messages frauduleux. L'intelligence artificielle **(IA)** et surtout le Machine Learning est une solution prometteuse pour classer ces messages en deux catégories : les messages **SPAM** et les **HAM**. Cependant, cette approche de classification montre des performances insatisfaisantes en raison du faible taux de réussite de la classification des messages valides. Dans ce travail, nous allons expérimenter quelques modèles les plus utilisés pour la classification de texte ces dernières années et découvrir quel modèle est le meilleur pour résoudre ce problème.

Mots clés : Machine Learning, la Classification des message, Spam, Ham, L'intelligence artificielle.

Abstract

In recent decades, the use of email has become widespread, leading to spam or fraudulent messages. Artificial intelligence **(AI)** and especially machine learning is a promising solution to classify these messages into two categories: messages **SPAM** and **HAM**. However, this classification approach shows unsatisfactory performance due to the low success rate of valid message classification. In this work, we will experiment with the most used models for text classification in recent years and find out which model is the best to solve this problem.

Keywords: Machine Learning, message classification, Spam, Ham, Artificial intelligence.

ملخص

في العقود الأخيرة، انتشر استخدام البريد الإلكتروني على نطاق واسع، مما أدى إلى ظهور رسائل غير مرغوب فيها أو رسائل احتيالية. يعد الذكاء الاصطناعي **(AI)** وخاصة التعلم الآلي حلا واعدا لتصنيف هذه الرسائل إلى فئتين: الرسائل **SPAM** و **HAM** ومع ذلك يُظهر نهج التصنيف هذا أداء" غير مر ض بسبب انخفاض معدل نجاح تصنيف الرسائل ، الصحيحة.

في هذا العمل، سنقوم بتجربة الخوارزميات الأكثر استخداما لتصنيف النص في السنوات الأخيرة ومعرفة الخوارزمية الأفضل لحل هذه المشكلة .

الكلمات المفتاحية : التعلم الآلي ، تصنيف الرسائل ، الرسائل المرغوب فيها ،الرسائل الغير مرغوب فيها ، الذكاء الاصطناعي.

Table des matières

LISTE DES ABREVIATIONS

IA	Intelligence Artificielle
DL	Deep Learning
LSTM	Long Short Term Memory
CNN	Convolutional Neural Network
ANN	Artificiel Neural Network
RNN	Recurrent Neural Network..
ML	Machine Learning
SVM	Support Vector Machine
NB	Naïve Bayes
KNN	K Nearest Neighbor
DT	Decision Tree

LISTE DES FIGURES

Table des tableaux

Introduction générale

1 Introduction générale

Au cours des dernières décennies, le gonflement des données textuelles sur Internet et les médias sociaux ont vu une explosion de l'utilisation du courrier électronique, entraînant l'envoi de milliers, voire de millions de messages.

Cependant, cette croissance a également causé des problèmes tels que l'apparition des messages malveillants, indésirables, ou ce qu'on appelle aussi : **spam**.

Pour résoudre ce problème, l'intelligence artificielle (IA), en particulier l'apprentissage automatique, offre une solution prometteuse. Les modèles d'apprentissage automatique peuvent classer ces messages en deux catégories distinctes : **SPAM** et **HAM**.

Dans ce travail on va présenter une étude concernant des méthodes de classification issues de l'apprentissage automatique Cette étude est appliqué sur une base de test de SPAM dans laquelle on va essayer différentes méthodes utilisant des distributions de probabilités différentes : Distribution normale et distribution de Rice.

Ce mémoire est organisé comme suit :

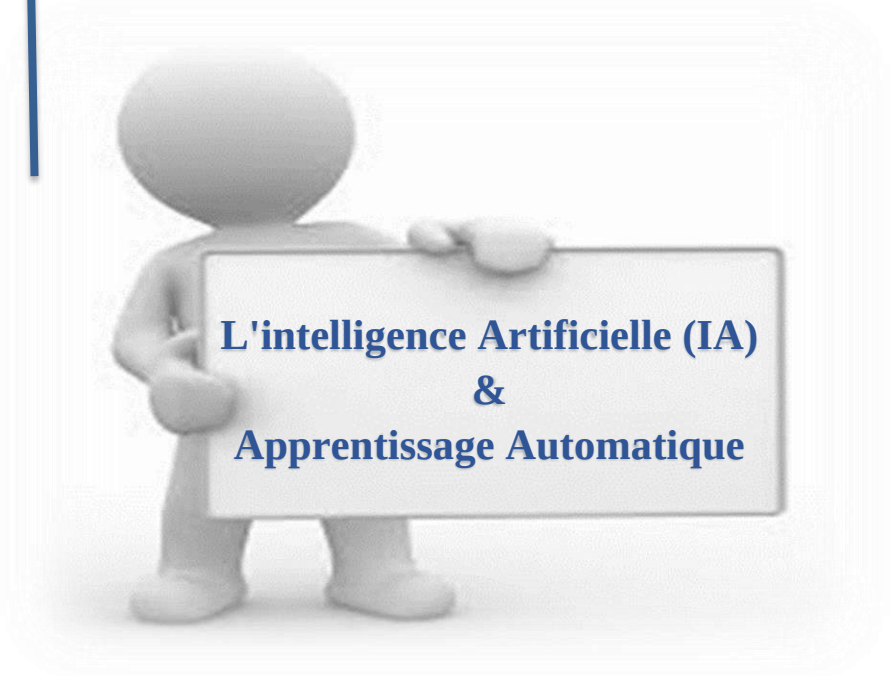
Chapitre 1 : Commence par l'Intelligence Artificielle (IA) & Apprentissage Automatique.

Chapitre 2 : Nous présentons la Sécurite Réseaux, les différents attaques ainsi la sécurité de la messagerie électronique, la définition des spam, leurs objectifs, leurs subjectivités, leurs vecteurs et les différentes techniques utilisées pour détecter ce type de courriel.

Chapitre 3 : Ce chapitre décrit la vu technique IA pour la detection du spam, l'approche que nous proposons et explique la méthodologie utilisée.

Chapitre 4 : Ce dernier chapitre est principalement consacré à la conduite et à la réalisation d'expériences. Il définit les outils et le langage de programmation utilisés dans leur mise en œuvre, et fournit une analyse détaillée des résultats obtenus. On termine notre travail par une conclusion.

Chapitre I


L'intelligence Artificielle (IA)
&
Apprentissage Automatique

2 Chapitre I : l'Intelligence Artificielle (IA) et Apprentissage Automatique

2.1 Introduction

L'apprentissage automatique est une branche de l'intelligence artificielle (IA) et de l'informatique qui porte sur l'utilisation des données et des algorithmes pour imiter la manière dont les êtres humains apprennent, afin d'améliorer progressivement sa précision.
Nous considérerons que l'intelligence est une faculté qui permet de résoudre des problèmes et de surmonter des obstacles, en mobilisant des connaissances, soit acquises, soit à construire [1].

2.2 L'Intelligence Artificielle (IA)

Selon le Larousse, l'intelligence Artificielle se définirait comme étant « l'ensemble de théories et de techniques mises en œuvre en vue de réaliser des machines capables de simuler l'intelligence ».
Ce serait, de ce fait, des ordinateurs ou des machines dotées de programmes capables de performances similaires à l'intelligence humaine, ou même, amplifiées par la technologie. Ces machines sont en mesure de :

- Raisonner
- Traiter de grandes quantités de données
- Discerner des modèles indétectables par l'œil d'un humain
- Comprendre et analyser ces modèles
- Interagir avec l'Homme
- Apprendre progressivement
- Améliorer continuellement ses performances.

Depuis 1950, année à laquelle l'IA a été créée, cette dernière est en mutation. Elle a, même en janvier 2018, franchi l'étape selon laquelle elle dépasserait l'intelligence humaine.

Selon Harry Shum, Président Exécutif de Microsoft, l'IA fonctionne seulement s'il y a présence « d'une vaste quantité de data; d'une puissance informatique extraordinaire, notamment grâce au cloud; et des algorithmes révolutionnaires, basés sur le deep learning».
L'IA s'applique aujourd'hui dans des domaines variés tels que :

- Les jeux de réflexion
- La recherche mathématique
- La finance
- La médecine
- Les assistants personnels et la domotique
- La reconnaissance faciale et la compréhension des langues
- La robotique.

2.3 Apprentissage Automatique

L'apprentissage automatique classique est souvent classé en fonction de la façon dont un algorithme apprend à devenir plus précis dans ses prédictions.
Il existe deux principaux types : **l'apprentissage supervisé, l'apprentissage non supervisé.**

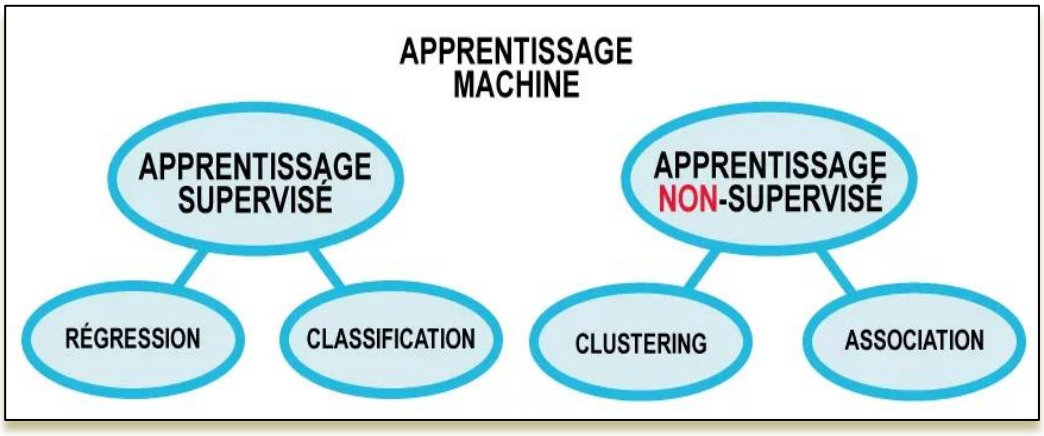


Figure 1-1 : Principaux types d'apprentissage

2.3.1 Apprentissage Supervise

L'apprentissage supervisé est une sous-catégorie de l'apprentissage automatique et de l'intelligence artificielle. Il est défini par son utilisation d'ensembles de données étiquetés pour former des algorithmes qui permettent de classer les données ou de prédire les résultats avec précision. L'objectif est de donner un sens aux données dans le contexte d'une question spécifique. L'apprentissage supervisé est utilisé pour des problèmes de classification et de régression, comme la détermination de la catégorie à laquelle appartient un article de presse, ou la prévision du volume des ventes pour une date future donnée.

L'apprentissage supervisé est fait en utilisant une vérité, c'est-à-dire qu'on a une connaissance préalable de ce que les valeurs de sortie pour nos échantillons devraient être. Par conséquent, le but de ce type d'apprentissage est d'apprendre une fonction qui, compte tenu d'un échantillon de données et de résultats souhaités, se rapproche le mieux de la relation entre les entrées et les sorties observables dans les données.

Dans l'apprentissage supervisé, on a deux types d'algorithmes :

Les algorithmes de **régression**, qui cherchent à prédire une valeur continue, une quantité.
Les algorithmes de **classification**, qui cherchent à prédire une classe/catégorie.

Pour créer un modèle d'apprentissage supervisé, on peut recourir à différents algorithmes , on peut citer en guise d'exemple la régression linéaire et logistique, **l'arbre de décision** avec différentes variables de sortie, le **Naive Bayes**, **Random Forest**, **SVM** et **k-NN**.

2.3.1.1 L'arbre de décision

est l'un des premiers algorithmes de Machine Learning que les Data Scientists apprennent au cours de leur formation. Il est utilisé pour représenter visuellement et explicitement les décisions et la prise de décision pour des problèmes de classification ainsi que pour des problèmes de régression. Il représente aussi l'élément de base de plusieurs modèles comme le Random Forest ou XGBoost.

Tout d'abord, **qu'est-ce qu'un arbre de décision ?** Visuellement, cela ressemble à une structure descendante composée de **nœuds** : chaque nœud possède une condition qui amène à plusieurs réponses, ce qui dirige à un prochain nœud.

Sur ce graphique, chaque nœud peut avoir aucune ou plusieurs possibilités : cela va dépendre de s'il est terminal ou non.
Prenons l'arbre de décision suivant qui indique si l'on doit prendre un parapluie avec nous en fonction du temps.

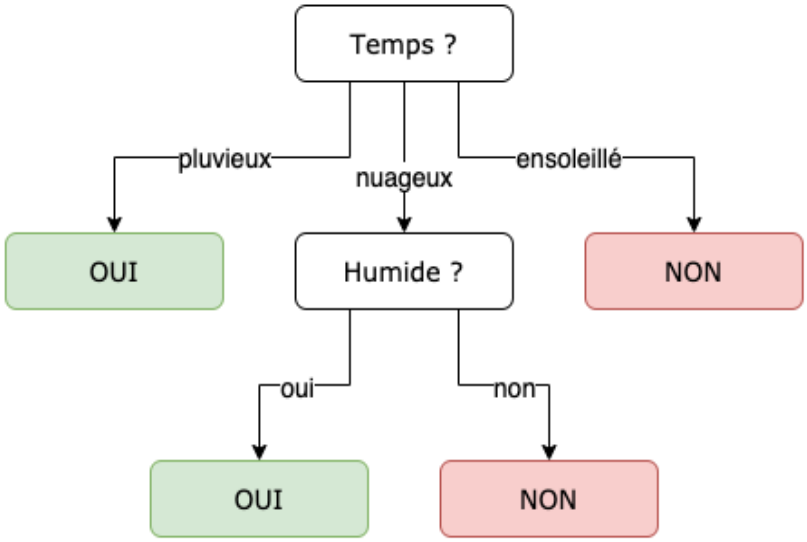


Figure 1-2 : l'arbre de décision

Dans cet exemple, un jour ensoleillé donnera directement la réponse **NON**, alors qu'un jour nuageux donnera la réponse **OUI** ou **NON** en fonction de l'humidité.

Sur ce graphique, chaque nœud peut avoir aucune ou plusieurs possibilités : cela va dépendre de s'il est terminal ou non.
En bref, l'arbre de décision est un outil élémentaire et indispensable de machine Learning qu'il vous faut maîtriser. Il est à la fois simple et lisible mais il constitue une méthode efficace de prise de décision.

2.3.1.2 Le Random Forest (pour forêt aléatoire)

est un algorithme de Machine Learning très populaire auprès des Data Scientists en raison de sa précision, de sa simplicité et de sa flexibilité. Cet algorithme peut être utilisé pour résoudre les problèmes de régression et de classification. Il est fréquemment adopté dans de nombreux domaines tels que les banques et le commerce en ligne pour prédire des comportements et des résultats futurs. Nous allons détailler ensemble le principe de fonctionnement de cet algorithme, déterminer en quoi est-il avantageux par rapport aux autres modèles de Machine Learning, et surtout, dans quelles situations il est préférable de l'utiliser.
Le Random Forest est un ensemble d'arbres de décision utilisés pour prédire une quantité ou une probabilité. Passons rapidement en revue les arbres de décision, car ce sont les éléments de base du modèle de forêt aléatoire.

2.3.1.3 k-NN (pour K-nearest neighbors) Algorithme des K-ppv (le plus proche voisin)

C'est une approche très simple et directe. Elle ne nécessite pas d'apprentissage mais simplement le stockage des données d'apprentissage. Son principe est le suivant. Une donnée de classe inconnue est comparée à toutes les données stockées. On choisit pour la nouvelle donnée la classe majoritaire parmi ses K plus proches voisins (Elle peut donc être lourde pour des grandes bases de données) au sens d'une distance choisie. Quelle distance Afin de trouver les K plus proches d'une donnée à classer, on peut choisir la distance euclidienne. Soient deux données représentées par deux vecteurs xi et xj, la distance entre ces deux données est donnée par :

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\sum_{k=1}^{d} (x_{ik} - x_{jk})^2}$$

2.3.1.4 SVM (Support Vector Machine) SVM (Support Vector Machine ou Machine à vecteurs de support)

est un algorithme d'apprentissage automatique supervisé qui peut être utilisé pour les problèmes de classification ou de régression. Toutefois, il est surtout utilisé dans les problèmes de classification.

Il était extrêmement populaire à l'époque où il a été développé, dans les années 1990, et continue de l'être car il produit une précision significative avec un minimum de puissance de calcul.
Le SVM joue aussi un rôle important dans la reconnaissance des modèles qui est l'un des domaines de recherche les plus populaires et actif de nos jours. Dans cet article, nous allons détailler les différents concepts du SVM ainsi que préciser la manière optimale pour l'utiliser.

Le principe des SVM consiste à ramener un problème de classification ou de discrimination à un hyperplan (*feature space*) dans lequel les données sont séparées en plusieurs classes dont la frontière est la plus éloignée possible des points de données (ou *marge maximale).*

On prendra un exemple simple de classification pour mieux comprendre de quoi il s'agit. On dispose d'une population **composée de 50% de femme et 50% d'hommes**. En utilisant un échantillon de cette population, on veut créer un ensemble de règles qui nous guideront dans la **classification de sexe** pour le reste de la population.

On suppose que les deux facteurs de différenciation identifiés sont : la taille de l'individu et la longueur des cheveux. [01].

Voici un diagramme de dispersion de l'échantillon :

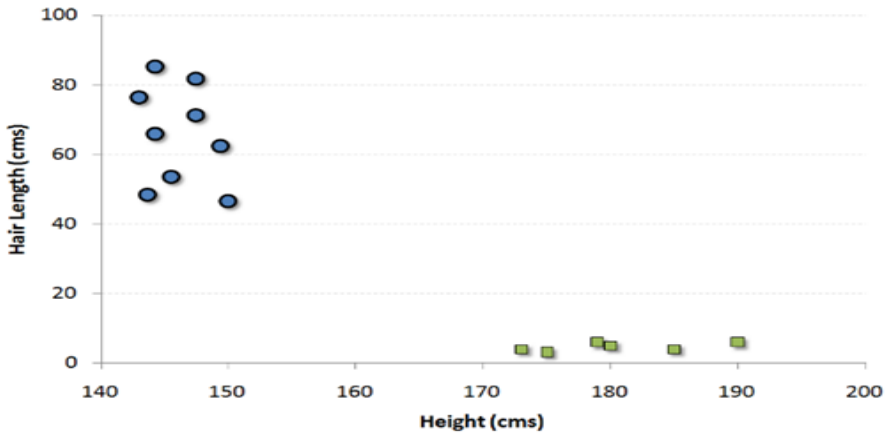


Figure 1-3 : Diagramme de dispersion de l'échantillon

Les cercles bleus dans le graphique représentent les femmes et les carrés verts représentent les hommes. Visuellement, on peut distinguer ces deux classes comme suit :

Les hommes de cette population ont une taille moyenne plus élevée.
Les femmes de cette population ont des cheveux plus longs.
À l'aide de l'apprentissage supervisé, le modèle peut prédire les résultats **sur la base d'expériences antérieures**. Mais cela ne veut pas dire qu'il est capable de gérer des tâches complexes. En effet, ce genre d'algorithmes ne sont **pas capables de fournir de bons résultats si les données de test sont différentes de ceux de l'entrainement**, et ils ne peuvent fonctionner correctement que si on possède une bonne connaissance des classes des éléments du jeu de données.
Il existe de nombreuses applications de l'apprentissage supervisé :

- Étude de clientèle.
- Diagnostic médical.
- Reconnaissance vocale.
- Vision par ordinateur.
- Traitement automatique du langage (NLP).
- On peut l'utiliser également pour la détection de spams dans les mails, la détection de fraude, la gestion des chatbots ainsi que pour la robotique.

2.3.2 Apprentissage Non Supervise

Les algorithmes d'apprentissage automatique non supervisés sont utilisés lorsque l'information utilisée pour entraîner le modèle n'est ni classifiée ni étiquetée. Le modèle en question étudie ses données d'entrainement dans le but de déduire une fonction pour décrire une structure cachée à partir ces données. À aucun moment le système ne connaît la sortie correcte avec certitude. Au lieu de cela, il tire des inférences des ensembles de données quant à ce que la sortie devrait être.

- **Clustering :** L'une des applications les plus courantes de l'apprentissage non supervisé est le regroupement. Le clustering est le processus de regroupement des objets similaires de manière à ce que les objets dans le même groupe (appelés cluster) soient plus similaires que ceux des autres groupes.

- **Détection d'anomalies :** Une autre application de l'apprentissage non supervisé est la détection des anomalies. La détection des anomalies est le processus d'identification des événements rares ou des anomalies dans les données.

- **Réduction de la dimensionnalité :** l'apprentissage non supervisé peut également être utilisé pour la réduction de la dimensionnalité. La réduction de la dimensionnalité est le processus de réduction du nombre de variables dans un ensemble de données tout en conservant autant d'informations que possible. Ceci est utile dans les situations où l'ensemble de données a un grand nombre de variables, ce qui rend difficile l'analyse.

- **Minage des règles d'association :** L'extraction des règles d'association est une autre application de l'apprentissage non supervisé. L'exploitation des règles d'association est le processus de découverte de relations entre les variables dans un ensemble de données.

Les données bruitées sont encore l'un des problèmes d'apprentissage automatique les plus courants des Data Scientists. Heureusement, nous avons maintenant accès à un large éventail de technologies et de techniques qui permettent de résoudre plus efficacement les problèmes de compression des données : un auto-encodeur est l'un d'entre eux.
Un auto-encodeur est une structure de réseaux neuronaux profonds qui s'entraîne pour réduire la quantité de données nécessaires pour représenter une donnée d'entrée. Ils sont couramment utilisés en apprentissage automatique pour effectuer des tâches de compression de données, d'apprentissage de représentations et de détection de motifs [01].

Les auto-encodeurs sont des réseaux de neurones un peu particuliers, qui possèdent exactement le même nombre de neurones sur leur couche d'entrée et leur couche de sortie. Le but d'un auto-encodeur est d'avoir une sortie la plus proche possible de l'entrée. L'apprentissage est donc auto-supervisé car la perte à minimiser est le coût de reconstruction entre la sortie et l'entrée. Les données n'ont ainsi pas à être labellisées, parce qu'elles sont leurs propres labels, ce qui fait alors de ce modèle un modèle non supervisé. Supposons que nous avons une image 100 x 100 avec laquelle nous voulons alimenter un encodeur pour réduire ses dimensions de manière appropriée. Sur l'espace de dimension de 10000 pixels, disons seulement 1000 de ces composants sont des données contenant les informations les plus utiles et décisives, en d'autres termes, des données représentatives de cette image. L'espace dimensionnel latent de l'encodeur automatique sera constitué de cet espace de dimensions inférieures avec les informations les plus utiles pour la reconstruction [01].

Conclusion

Dans ce chapitre nous avons présenté le Machine Learning qui est un des champs de l'Intelligence Artificielle (**IA**) qui consiste en l'automatisation de l'apprentissage d'un algorithme, notamment par l'analyse, la sélection et le traitement de données. Nous avons donné un bref historique sur l'apprentissage automatique, ces différents types (non supervisé, Supervisé).

Chapitre II

3 CHAPITRE II : Securite Réseaux

3.1 Introduction

Les systèmes informatiques sont au cœur des systèmes d'information Ils sont devenus la cible de ceux qui convoitent l'information.

Assurer la sécurité de l'information implique d'assurer la sécurité des systèmes informatiques.

3.2 Sécurité informatique :

Les progrès technologiques, le développement des moyens de communication, l'ouverture du monde aux nouvelles technologies et le transfert de divers types de données sur des réseaux, les rendant vulnérables aux menaces à la sécurité émanant de personnes non autorisées ou de concurrents. La protection des informations est devenue une nécessité pour les particuliers ou les entreprises. Pour ce faire, elle utilise des techniques et des mécanismes d'authentification et un contrôle de l'accès. Des outils doivent être utilisés pour garantir la sécurité des informations, notamment Firewall, VPN, IDS…etc. Le système de détection d'intrusion (IDS) est l'un de ces outils cruciaux des opérations de défense, La détection d'intrusion consiste essentiellement à rechercher des signes d'attaque. Lorsqu'une intrusion est détectée, le système de détection d'intrusion peut prendre Les procédures nécessaire Selon son type et sa programmation par l'administrateur.

La technologie de sécurité:

C'est l'ensemble de technologies utilisées pour réduire les vulnérabilités du système d'information contre les attaques accidentelles ou intentionnelles et ça c'est le but d'utiliser la sécurité pour le système d'information. La sécurité informatique est caractérisée généralement par les cinq propriétés ou objectifs suivants :

Disponibilité : Lorsqu'un utilisateur du système d'information demande des informations, la ressource doit être disponible pour répondre aux personnes autorisées uniquement.

Confidentialité : assurer que les informations sont cachées sur le système afin qu'elles ne puissent être lue que par les personnes autorisées.

Intégrité : Garantit l'impossibilité de modifier les informations relatives au système par des individus non autorisés sans l'intervention des personnes autorisées sans avoir les informer.
Non-répudiation : Est de prouver la source de données pour que les individus ne puissent pas nier leur participation à la communication.

L'authentification : Est d'assurer l'identité de l'utilisateur, dans le sens que doit être garantir l'identité de chacune des parties impliquées dans la communication, aussi il faut également assurer le contrôle d'accès aux ressources pour les individus autorisés (l'accès au compte e-mail avec une adresse et mot de passe correcte).

3.3 Système d'information :

Le système d'information est l'ensemble des moyens technologiques, organisationnels et humains permettant de collecter, stocker, traiter et distribuer de l'information entre les organisations.

3.3.1 Sécurité des systèmes d'information :

La sécurité des systèmes d'information est atteinte lorsque les objectifs de sécurité tel que la disponibilité, l'intégrité, la confidentialité, l'authentification et la non-répudiation sont garantis d'être réalisé, et confirmer que les méthodes, techniques et outils nécessaires à la protection des ressources du système d'information sont utilisés.

3.3.2 Domaines de la sécurité :

L'informatique étant l'un des fondements de l'entreprise, qui intervient également dans tous les domaines, c'est pour ça il est nécessaire de veiller à la sécurité des systèmes d'information. Selon leur domaine d'application, les moyens de la sécurité se classifient en:

- Sécurité physique ;
- Sécurité de l'exploitation ;
- Sécurité logique ;
- Sécurité applicative ;
- Sécurité des télécommunications.

On retrouve aussi dans le domaine de la sécurité informatique l'usage de quelques termes qu'il faut les reconnaitre, parmi eux :

Vulnérabilité : Une faute dans le système d'exploitation créée durant son développement ou c'est une faiblesse dans la sécurité des systèmes d'informations ou des réseaux, etc., qui peut être exploitée sous forme d'une menace sur la sécurité est utilisée pour pénétrer les systèmes et les réseaux.

Intrusion : Action malveillante résultant d'une attaque externe qui a réussi à exploiter une vulnérabilité pour permettre à l'attaquant de contrôler le système ou le réseau.

Menace : possibilités et probabilités d'attaque contre la sécurité. Une menace est définie par le processus d'attaque, par la cible et par le résultat (conséquences de la réussite d'une attaque).

Attaque : C'est n'importe quelle action qui a le but de menacer la sécurité des informations et de nuire au moins à l'une des propriétés de la sécurité informatique (disponibilité, Confidentialité, Intégrité, L'authentification). Il s'agit d'une tentative d'intrusion, nous abordons dans ce qui suit les différents buts et classes de ces attaques (tentatives d'intrusion).

3.4 Réseaux informatiques

Les Réseaux informatiques sont nés du besoin de faire communiquer des terminaux distants avec un site central puis des ordinateurs entre eux. Dans un premier temps ces communications étaient juste destinées aux transports de données informatiques alors qu'aujourd'hui on se dirige plutôt vers des réseaux qui intègrent à la fois des données mais en plus, la parole, et la vidéo.

Station de travail : On appelle station de travail toute machine capable d'envoyer des données vers les réseaux (PC, MAC, SUN Terminal X, …).

Nœud : C'est une station de travail, une imprimante, un serveur ou toute entité pouvant être adressée par un numéro unique.

Serveur : Dépositaire centrale d'une fonction spécifique : service de base de donnée, de calcul, de fichier, mail, ….

Paquet : C'est la plus petite unité d'information pouvant être envoyer sur le réseau. Un paquet contient en général l'adresse de l'émetteur, l'adresse du récepteur et les données à transmettre.

Topologie : Organisation physique et logique d'un réseau. L'organisation physique concerne la façon dont les machines sont connectés **(Bus, Anneau, Étoile ….).**
La topologie logique montre comment les informations circulent sur les réseaux **(diffusion ou point à point)**.

Réseaux Homogènes : Tous les ordinateurs sont de mêmes constructeurs : **Aple-Talk**
Réseaux Hétérogènes : Les ordinateurs reliés au réseau sont de constructeurs divers : **Ethernet.**

3.5 Définition d'un protocole

Dans le domaine des réseaux, un protocole est un ensemble de règles permettant de formater et de traiter les données. Les protocoles de réseau sont comme un langage commun pour les ordinateurs. Les ordinateurs d'un réseau peuvent utiliser des logiciels et du matériel très différents, mais l'utilisation de protocoles leur permet de communiquer entre eux.

Les protocoles normalisés sont comme un langage commun que les ordinateurs peuvent utiliser, de la même manière que deux personnes de différentes régions du monde peuvent ne pas comprendre la langue maternelle de l'autre, mais elles peuvent communiquer en utilisant une troisième langue commune. Si un ordinateur utilise le **protocole Internet (IP)** et qu'un deuxième ordinateur le fait aussi, ils pourront communiquer, tout comme les Nations unies s'appuient sur leurs 6 langues officielles pour communiquer entre représentants du monde entier. Mais si un ordinateur utilise **IP** et que l'autre ne connaît pas ce protocole, ils ne pourront pas communiquer.

Sur Internet, il existe différents protocoles pour différents types de processus. Les protocoles sont souvent discutés en fonction de la couche du modèle **OSI** à laquelle ils appartiennent.
Le modèle **OSI** (Open Systems Interconnection) est une représentation abstraite du fonctionnement de l'Internet. Il contient **7** couches, chaque couche représentant une catégorie différente de fonctions de mise en réseau.

Les protocoles rendent ces fonctions de mise en réseau possibles. Par exemple, le protocole Internet **(IP)** est responsable de l'acheminement de données en indiquant d'où viennent paquets de données et quelle est leur destination. L'**IP** rend possible les communications de réseau à réseau. Par conséquent, **IP** est considéré comme un protocole de la **couche réseau (couche 3)**.

Autre exemple, le protocole de contrôle de transmission **(TCP)** assure le bon déroulement du transport des paquets de données sur les réseaux. Par conséquent, le **TCP** est considéré comme un protocole de **la couche transport (couche 4)**.

Comme décrit ci-dessus, **IP** est un protocole de la couche réseau responsable du routage. Mais ce n'est pas le seul protocole de la couche réseau.

IPsec : Internet Protocol Security **(IPsec)** établit des connexions **IP** chiffrées et authentifiées sur un réseau privé virtuel **(VPN)**. Techniquement, **IPsec** n'est pas un protocole, mais plutôt un ensemble de protocoles comprenant le protocole de sécurité d'encapsulation **(ESP)**, l'en-tête d'authentification **(AH)** et les associations de sécurité **(SA)**.

ICMP : Le protocole de message de contrôle Internet **(ICMP)** signale les erreurs et fournit des mises à jour d'état. Par exemple, si un routeur n'est pas en mesure de livrer un paquet, il renvoie un message **ICMP** à la source du paquet.

IGMP : Le protocole de gestion des groupes Internet **(IGMP)** établit des connexions réseau de type un à plusieurs. **IGMP** aide à mettre en place la multidiffusion, ce qui signifie que plusieurs ordinateurs peuvent recevoir des paquets de données dirigés vers une adresse **IP**.

3.5.1 Les protocoles utilisés sur Internet

Voici quelques-uns des protocoles les plus importants à connaître :

TCP : Comme décrit ci-dessus, TCP est un protocole de couche de transport qui assure une livraison fiable des données. TCP est destiné à être utilisé avec IP, et les deux protocoles sont souvent référencés ensemble sous le nom de TCP/IP.

HTTP : Le **protocole de transfert hypertexte (HTTP)** est la base du World Wide Web, l'Internet avec lequel la plupart des utilisateurs interagissent. Il est utilisé pour transférer des données entre des périphériques. HTTP appartient à la **couche application (couche 7)**, car il met les données dans un format que les applications (par exemple, un navigateur) peuvent utiliser directement, sans autre interprétation. Les couches inférieures du modèle OSI sont gérées par le système d'exploitation d'un ordinateur, et non par les applications.

HTTPS : Le problème avec HTTP est qu'il n'est pas chiffré - Tout attaquant qui intercepte un message HTTP peut le lire. **HTTPS** (HTTP Secure) corrige cela en chiffrant les messages HTTP.

TLS/SSL : **Transport Layer Security (TLS)** est le protocole que HTTPS utilise pour le chiffrement. TLS s'appelait auparavant **Secure Sockets Layer (SSL)**.

UDP : le **protocole UDP** (User Datagram Protocol) constitue une alternative plus rapide, mais moins fiable, au protocole TCP au niveau de la couche de transport. Il est souvent employé pour les services de type **diffusion vidéo** et jeux vidéo, pour lesquels une diffusion rapide des données est primordiale.

3.5.2 Les protocoles utilisés par les routeurs

Les routeurs de réseau utilisent certains protocoles pour découvrir les chemins de réseau les plus efficaces vers d'autres routeurs. Ces protocoles ne sont pas utilisés pour transférer des données utilisateur. Les principaux protocoles de routage réseau sont les suivants :

BGP : Le **protocole BGP (Border Gateway Protocol)** est un protocole de couche application que les réseaux utilisent pour diffuser les adresses IP qu'ils contrôlent. Ces informations permettent aux routeurs de décider par quels réseaux les paquets de données doivent passer pour atteindre leur destination.

EIGRP : Le protocole EIGRP (Enhanced Interior Gateway Routing Protocol) identifie les distances entre les routeurs. EIGRP met automatiquement à jour l'enregistrement des meilleures routes de chaque routeur (appelé table de routage) et diffuse ces mises à jour aux autres routeurs du réseau.

OSPF : Le protocole Open Shortest Path First (OSPF) calcule les itinéraires réseau les plus efficaces en fonction de divers facteurs, notamment la distance et la bande passante.

RIP : Le Routing Information Protocol (RIP) est un ancien protocole de routage qui identifie les distances entre les routeurs. RIP est un protocole de couche d'application.

3.5.3 Les protocoles utilisés dans les cyberattaques

Comme pour tout autre aspect de l'informatique, les attaquants peuvent exploiter le fonctionnement des protocoles de réseau pour compromettre ou submerger les systèmes. Nombre de ces protocoles sont utilisés dans les attaques par déni de service distribué (DDoS).

Par exemple, dans une **attaque SYN flood**, un attaquant tire parti du fonctionnement du protocole TCP. Il envoie des paquets SYN pour initier de manière répétée une **poignée de main TCP**, avec un serveur, jusqu'à ce que le serveur soit incapable de fournir un service aux utilisateurs légitimes parce que ses ressources sont bloquées par toutes les fausses connexions TCP.

Cloudflare offre un certain nombre de solutions pour stopper ces attaques et d'autres cyberattaques. **Cloudflare Magic Transit** est capable d'atténuer les attaques au niveau des couches 3, 4 et 7 du modèle OSI. Dans le cas d'une attaque SYN flood, Cloudflare gère le processus de poignée de main TCP pour le compte du serveur afin que les ressources du serveur ne soient jamais submergées par des connexions TCP ouvertes.

3.6 Sécurité réseaux

Les systèmes informatiques utilisent différents composants, allant de l'électricité aux machines en fonctionnement, en passant par les programmes exécutés sur le système d'exploitation et utilisant le réseau. Des attaques peuvent se produire dans chaque lien vers cette chaîne, s'il existe une faille pouvant être exploitée. Pour l'aspect technique, on définit que l'attaque c'est une exploitation d'une faille pour des fins illégales.

3.7 Techniques d'attaque

Il existe cinq formes d'attaque que nous détaillons comme suit :

3.7.1 L'attaque passive :

Les attaques passives sont toute action nous permettant d'analyser et de déchiffrer le trafic, de surveiller les communications et de capturer des informations d'authentification. L'attaquant utilise ce type d'attaque pour obtenir des informations ou de données facilement et à l'insu de la victime en interceptant les mots de passe, les numéros de carte de crédit et les emails.

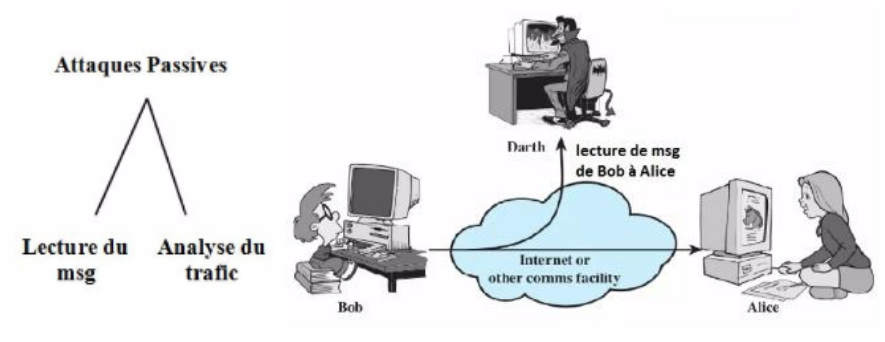


Figure 2-1 : L'attaque passive (L'analyse du trafic réseau).

3.7.2 L'attaque active :

Les attaques actives incluent les tentatives visant à contourner ou casser des fonctionnalités de sécurité afin de falsifier ou de voler des informations en insérant un code malveillant dans le système d'exploitation ou le réseau, ainsi que des menaces d'attaques actives visant à détecter ou à publier des fichiers de données, à refuser le service ou à modifier les données. Figure 2 : L'attaque active (rejeu)

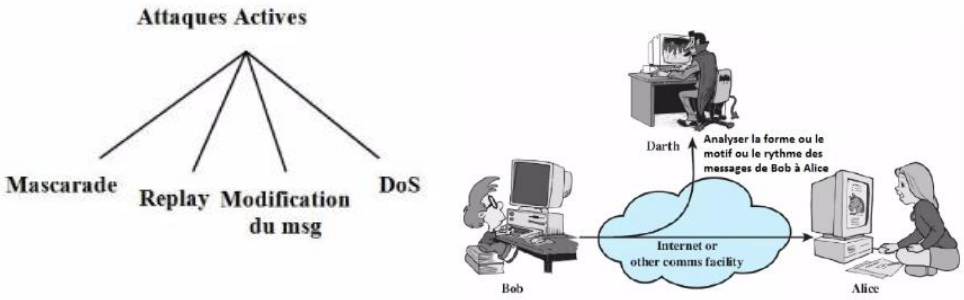


Figure 2-2 : L'attaque active (L'analyse du trafic réseau).

3.7.3 L'attaque externe :

L'attaquant doit utiliser la proximité physique pour pouvoir se connecter à des systèmes ou des réseaux via un accès secret ou ouvert afin de modifier, comparer et gérer l'accès aux informations.
L'attaque interne :

Dans ces cas d'attaques, l'attaquant peut faire partie de l'entreprise ou utiliser l'ingénierie sociale avec les personnes impliquées à la suite d'abus, de négligence ou de manque de connaissances. Dans les deux cas, ces attaques essayent d'espionner, de voler ou de détruire des informations, de les utiliser frauduleusement ou d'empêcher l'accès à d'autres utilisateurs autorisés.
L'attaque de distribution :

les attaques de distribution représentent toute modification malveillante du matériel ou du logiciel en usine ou lors de la distribution.

Ces attaques consistent à introduire un code malveillant dans un produit comme un port dérobé pour obtenir un accès non autorisé à des informations ou une fonction système.

Exemples des attaques :

S'il existe des failles de sécurité dans l'hôte ou le réseau, l'attaquant utilisera certainement de nombreuses attaques pour pouvoir exploiter les exploits de l'hôte ou du réseau, nous citons par exemple:

- **Craquage des mots de passe**.
- **Cheval de Troie** « Trojan Horse Un programme est caché dans un autre programme pour supprimer les soupçons, ce qui est dangereux si la victime installe le programme téléchargé, qui a portait un cheval de Troie qui va ouvrir une porte dérober au système pour certaines personnes qui utilise ce trojan.
- **IP spoofing** : L'attaquant change son adresse IP pour personnifiant l'identité d'un hôte confiant pour le permettre de bénéficier les privilèges de cet hôte afin d'accéder et manipuler avec les données critiques.
- **Les scans** : c'est la première et la plus importante étape de l'attaquant est d'obtenir suffisamment d'informations pour préparer une attaque plus efficace. Les informations pouvant être obtenues à partir de cette attaque sont le type de système d'exploitation du périphérique, les ports ouverts, etc.
- **Sniffing:** Il permet de surveiller et d'analyser le trafic réseau afin d'obtenir des informations pertinentes pour que l'attaquant peut avoir une bonne modélisation des attaques ultérieures.
- **Les attaques de déni de service** (denial of service) ont pour but de paralyser le serveur cible pour qu'il devienne inaccessible, au moins pour une durée de temps. De très nombreuses techniques existent pour épuiser les ressources d'un hôte cible, par exemple : ICMP Flooding, smurf, SYN flood, etc.
- **Etc.**

3.8 Techniques de sécurité :

C'est l'ensemble de procédures ou dispositifs qui sont conçu pour détecter, prévenir ou récupérer les attaques qui menacent la sécurité informatique, il existe plusieurs outils de prévention contre-attaques informatiques, Nous avons cité ci-dessous quelques mécanismes :

- **La protection physique :** avent de parler sur la sécurité des systèmes d'information premièrement il faut assurer la sécurité des matériels informatique et leurs emplacements.

- **Chiffrement** : Les algorithmes utilisent des clés pour convertir les données afin d'obtenir une sécurité robuste. Leur sécurité dépend du niveau de sécurité des clés.

- **Signature numérique** : Un mécanisme pour assurer l'intégrité des données et également pour authentifier l'auteur du document.

- **Bourrage de trafic** : Mécanisme assurant la confidentialité des données sur le volume de trafic en cas d'interception par des attaquants.

- **Contrôle d'accès** : Vérifier l'authentification des utilisateurs et leurs autorisations d'accéder aux données et vérifier leurs privilèges.

- **Antivirus** : Logiciel censé protéger l'ordinateur contre les logiciels (ou fichiers potentiellement exécutables) néfastes. Ne protège pas contre un intrus qui emploie un logiciel légitime, ou contre un utilisateur légitime qui accède à une ressource alors qu'il n'est pas autorisé à le faire.

- **Le pare-feu** : Un élément (logiciel ou matériel) du réseau informatique contrôlant les communications qui le travers. Il a pour fonction de faire respecter la politique de sécurité du réseau, celle-ci définissant quelles sont les communications autorisées ou interdites. N'empêche pas un attaquant d'utiliser une connexion autorisée pour attaquer le système. Ne protège pas contre une attaque venant du réseau intérieur (qui ne le traverse pas).

- **Détection d'intrusion** : Identifiez une activité anormale ou suspecte sur le moniteur réseau. Ne pas détecter les accès incorrects mais autorisés par les utilisateurs légitimes. Le problème c'est comment minimiser les taux de faux positifs et de faux négatifs.

- **Journalisation ("logs")** : Enregistrement des activités de chaque acteur. Permet de constater que des attaques ont eu lieu, de les analyser et potentiellement de faire en sorte qu'elles ne se reproduisent pas.

- **Analyse des vulnérabilités ("Security audit")** : Identification des points de vulnérabilité du système. Ne détecte pas les attaques ayant déjà eu lieu, ou lorsqu'elles auront lieu.

3.9 Détection et filtrage des spams

Le spam est un grand problème pour les internautes. Les augmentations récentes du taux de spam ont causé une grande inquiétude parmi la communauté Internet. De nombreuses solutions avaient été suggérées pour résoudre le problème. Dans ce chapitre, nous présentons tout d'abord les débuts du spam, ses objectifs, ses contenus, ses impacts et les différentes techniques utilisées pour détecter ce type de courriels.

3.9.1 Naissance et débuts du spam

Origine du mot spam En 1937 La société Hormel Foods1 organise un concours pour trouver un nouveau nom pour leur jambon épicé, Ce nom doit être aussi caractéristique que le goût du produit « Spiced Ham » et qui propose « Spam » pour ce produit, fut donc la marque retenue. Cette viande précuite en boîte souvent synonyme de mauvaise nourriture a été largement utilisée par l'intendance des forces armées américaines pour la nourriture des soldats pendant la Seconde Guerre mondiale et sera introduite dans diverses régions du monde à cette occasion.

3.9.2 Définition du spam

Le spam est un message électronique non sollicité, envoyé massivement à un grand nombre de destinataires, à des fins publicitaires ou malveillantes. Le terme spam est aussi utilisé pour désigner le même type de message transmis par d'autres moyens de communication électroniques tels que les messageries instantanées, les blogs, les forums, et plus récemment, des réseaux de téléphonie mobile, via les SMS ou MMS. Même si le moyen de communication est différent, les techniques d'envoi et de détection restent relativement similaires. Le premier spam (**Figure 2.3**) date du 3 mai 1978. Ce jour-là, sur le réseau ARPANET (est le premier réseau à transfert de paquets développé aux États-Unis), Gary Thuerk, commercial de la société informatique DEC (**D**igital **E**quipment **C**orporation), invitait par e-mail 393 personnes.

Le message se présentait ainsi :

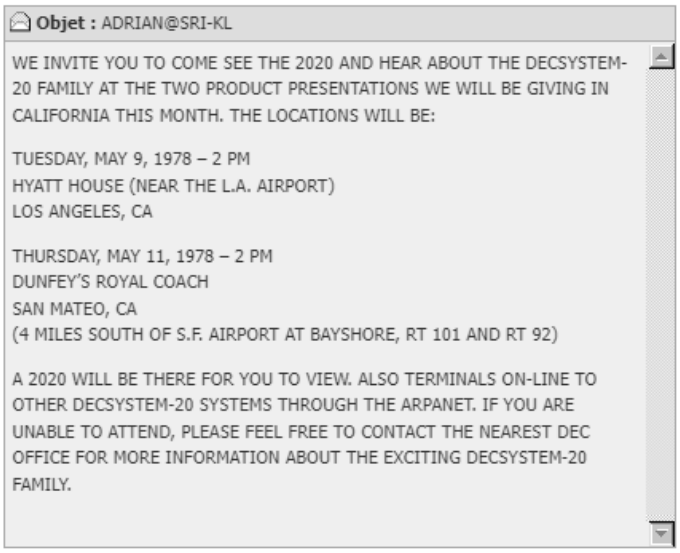


Figure 2-3 : Le premier spam

Ce message indésirable n'était hélas que le premier d'une longue série. Le spam était né.

3.10 Objectifs et statistiques sur les spam

Au départ, le spam visait principalement des objectifs publicitaires. Aujourd'hui, il s'est considérablement développé, diversifié et complexifié, pour atteindre de plus en plus souvent des objectifs malveillants. En effet, Le spam s'est non seulement développé en termes de volume, mais également en termes de contenu.

Aujourd'hui, les objectifs des spam sont très variés en voici une liste non exhaustive :

- **Hameçonnage (ou phishing) :** L'objectif est de réussir à se faire passer pour un organisme connu par l'utilisateur, dans le but de lui voler des informations à caractère confidentiel. Par exemple, on reçoit un mail provenant "apparemment" de notre banque, ou d'un autre site où l'on dispose d'informations personnelles.
  Dans ce mail, il est demandé de cliquer sur un lien (pour des motifs divers : réactualisation, etc.), après avoir cliqué sur ce lien, une page web s'affiche... sur laquelle il est demandé de rentrer ses coordonnées bancaires ou toute autre information personnelle. Parmi les sites Top les plus contrefaits pour les attaques de phishing, on retrouve eBay, Paypal et Bank of America.

- **Publicité :** L'objectif est de vanter les mérites d'un produit quelconque. Il s'agit par exemple de produits pharmaceutiques, de produits de luxe, de logiciels divers et variés, de jeux d'argent. Ils peuvent également soutenir-agate idées politiques, culturelles ou religieuses et / ou organisations.

- **Scam** : Il s'agit d'une attaque basée sur la naïveté des destinataires dans le but de leur soutirer de l'argent. L'exemple le plus courant est le scam nigérien : un dignitaire d'un pays d'afrique vous demande de servir d'intermédiaire pour une transaction financière importante, en vous promettant un bon pourcentage de la somme. Pour amorcer la transaction, il vous faut donner de l'argent.

- **Canular :** L'objectif est de faire circuler une information semblant très sensible, souvent avec un caractère d'urgence : fausse alerte de virus, fausse alerte de contamination potentielle, chaine de solidarité... Par exemple : « un nouveau virus très dangereux se propage, il faut faire circuler l'information » ; « des sous-vêtements sont infectés par une dangereuse bactérie ».

- **Malware :** Est un logiciel conçu pour infiltrer ou endommager un système informatique. Il est communément pris pour contenir des virus informatiques, vers, chevaux de Troie, spywares et adwares. Ce type de logiciel est souvent envoyé en tant que non suspect d'une pièce jointe. Lorsque l'utilisateur ouvre le fichier, le logiciel malveillant s'installe. L'interdépendance entre les spams et les logiciels malveillants a évolué Spam logiciels malveillants propagation des e-mails, les logiciels malveillants est utilisé pour infecter un hôte de sorte que l'hôte peut être contrôlé à distance et utilisé pour l'envoi de plus de spams. Ces hôtes infectés sont désignés comme des « ordinateurs zombies ». Beaucoup de gens croient que la plupart des spams sont envoyés par des botnets, qui constituent un réseau de PC zombies [10].

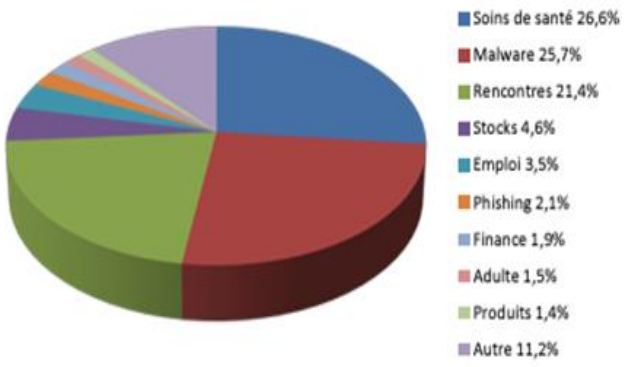


Figure 2-4 Répartition des spam par contenu

On a quelques statistiques sur le taux global de spam entre les années 2012 et 2017 présenté dans la **figure 2.5.**

Dans la dernière période il a été constaté que le spam représentait 55% de tous les messages électroniques, comme au cours de l'année précédente.

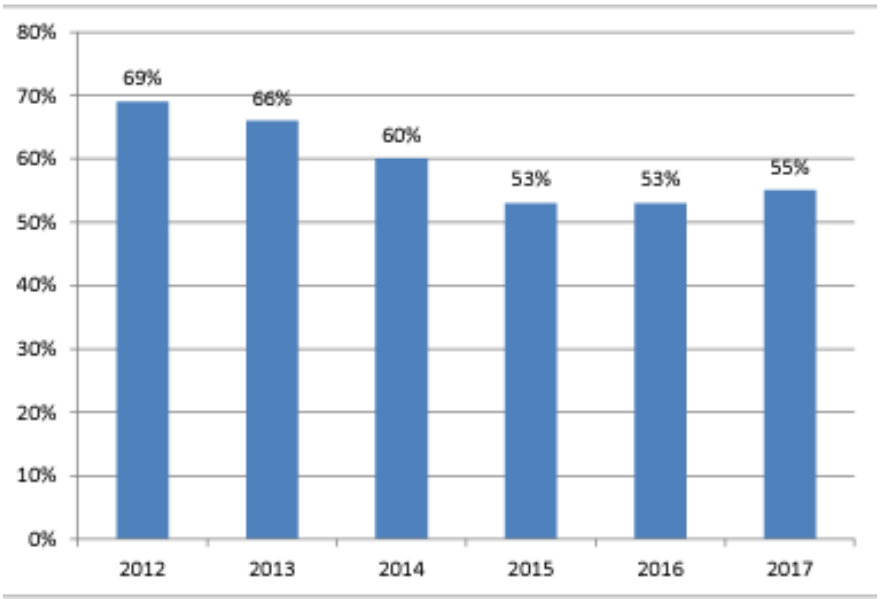


Figure 2-5 développement de spam en termes de volume

3.11 Impactes du spam sur les utilisateurs et les fournisseurs

Dans cette section, nous présentons les effets du spam, au niveau des utilisateurs, entreprises et FAI (Fournisseur d'Accès à Internet aussi appelé fournisseur de services Internet ou FSI) [9]

Perte de temps :

- Encombrement anormal des boîtes aux lettres.
- Suppression des courriels indésirables.
- Configuration et maintenance des filtres.
- Consultation des courriels rejetés pour y détecter les bons à cause du risque de passer à côté d'emails importants mal catalogués par les outils de détection anti-spam.

Perte de bande passante et d'espace disque :

- Spécialement pour les utilisateurs de modems.
- Les pièces jointes des virus et spam peuvent être grands.

Pertes financières non négligeables aux niveaux des entreprises et FAI :

- Une augmentation des coûts de gestion opérationnelle et support lié à la gestion anti spam.
- Perte de productivité des salariés, Selon une étude, le spam aurait coûté environ 712 $ par employé et par an aux entreprises. À ce chiffre, il faut rajouter 113 à 183 $ par employé et par an pour la gestion des emails en quarantaine.

3.12 Techniques de filtrage du spam

Plusieurs techniques de lutte contre le spam sont possibles et peuvent être cumulées : analyse statistique (filtre bayésien), filtrage par mots clés, listes blanches, listes noires. Ces techniques de lutte doivent s'adapter en permanence car de nouveaux types
de spam réussissent à les contourner.

Deux solutions de détection de spam sont envisageables :

La détection au niveau du serveur mail FAI et la détection au niveau de l'utilisateur final.

Ces outils peuvent être divisés en deux groupes :
le Filtrage d'Enveloppe, et **le Filtrage de Contenu**.

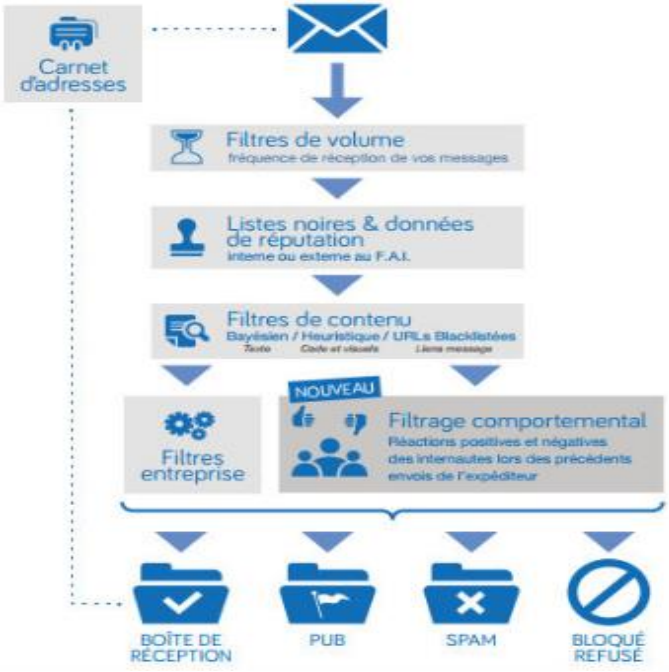


Figure 2-6 : filtrage de volume et filtre de contenu

3.12.1 Filtrage d'enveloppe

Ce type de filtrage s'applique uniquement sur l'en-tête (Elle constitue les informations de base de ce dernier : expéditeur, destinataire, date d'envoi, serveur source, Objet) du message, qui contient souvent assez d'informations pour pouvoir distinguer un spam.
Cette technique appliquée au niveau du serveur FAI présente l'avantage de pouvoir bloquer les courriels avant même que leur corps ne soit envoyé, ce qui diminue grandement le trafic sur la passerelle **SMTP**. Dans cette catégorie, nous trouvons les techniques suivantes :

3.12.1.1 Filtrage par listes noires :

Ces listes consistent à pré-déclarer une liste de « mauvais expéditeurs », (adresses emails, noms des domaines, pays, adresse IP), desquelles le destinataire refuse de recevoir des emails. Ces listes peuvent être :

- Créées par l'administrateur ou l'utilisateur.
- Téléchargées via le web (cela nécessite une mise à jour très régulière pour un filtrage optimisé)
- Consultées en temps réel sur le web (RBL, Real Time Blackhole List).

Pour contourner ces listes les spammeurs, changent très fréquemment leurs adresses d'expédition (email, ou IP). [10]

3.12.1.2 Filtrage par listes blanches :

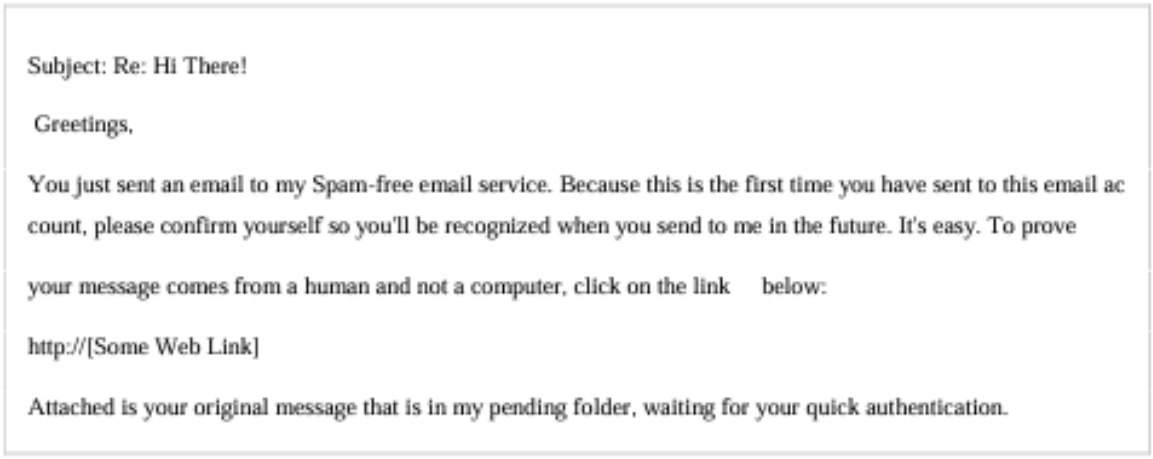
Ces listes consistent à pré-déclarer une liste de (adresses emails, noms des domaines, adresse IP) sûres desquels le destinataire accepte de recevoir des emails. Par défaut très peu d'hôtes sont considérés comme sûrs car leurs adresses pourraient être usurpées par les spammeurs. Tout comme la liste noire, la liste blanche a également besoin d'une mise à niveau continue et de rafraîchissement. [11]

3.12.1.3 Filtrage par liste grise :

La liste grise est un mixte entre la liste blanche et la liste noire. Ce qui se produit est qu'à chaque fois qu'une boîte aux lettres donnée reçoit un email d'un contact inconnu, cet email est suspendu avec un message de réponse automatique contenant un lien permettant de valider l'envoi. Ceci a pour but de détecter les robots, les spammeurs ne se rendront pas compte qu'ils doivent émettre une validation afin que le message soit accepté. Dans le cas d'un réel email attendu et que l'expéditeur n'est pas énuméré dans l'une ou l'autre des listes noire et blanche, alors il sera positionné en liste grise.

Si l'expéditeur satisfait la demande de confirmation (souvent un lien Web à cliquer), il obtiendra alors le passage à liste blanche et ses messages vous seront acheminés. C'est en fait l'ouverture dynamique de la liste blanche.

Par exemple : la figure suivante présente une réponse de cette technique.



Subject: Re: Hi There!

Greetings,

You just sent an email to my Spam-free email service. Because this is the first time you have sent to this email ac count, please confirm yourself so you'll be recognized when you send to me in the future. It's easy. To prove

your message comes from a human and not a computer, click on the link below:

http://[Some Web Link]

Attached is your original message that is in my pending folder, waiting for your quick authentication.

Figure 2-7 Exemple sur une réponse de technique de la liste grise [10]

3.12.1.4 Filtrage par vérification du domaine :

Les destinataires sont configurés de sorte qu'ils n'acceptent que les messages provenant de domaines spécifiques. Les e-mails dont les domaines ne sont pas mentionnés ne seront pas reçus. De cette façon, beaucoup de spam est bloqué. [12]

3.12.2 Filtrage du contenu :

Ce type de filtrage se fait au niveau de l'utilisateur où son contenu est analysé pour détecter les spam qui ont réussi à passer à travers le filtre d'enveloppe. Dans cette catégorie nous trouvons les techniques suivantes :

3.12.2.1 Filtrage par mots clés :

L'administrateur doit indiquer la liste des mots clés à détecter afin de déterminer qu'un mail est un Spam. Par exemple, tous les emails qui contiennent les mots : viagra, argent, money, drogue seront détectés comme Spam.
Ce filtre se base sur les mots clé inclus dans les mails. L'analyse est très rapide, mais peux efficace. Car cela demande un suivi manuel et les Spammeurs font varier les mots clé afin d'éviter ce filtre. Par exemple, on retrouve M.O.N.E.Y. ou encore m*o*n*e*y. [13]
Filtrage par caractères :
Il s'agit de bloquer les emails qui contiennent certains caractères ou police de caractère, ou certaines langues utilisées dans ces emails.

3.12.2.2 Filtrage d'image :

Il s'agit d'analyser les images obtenues dans les messages au niveau des propriétés du fichier image (format, taille du fichier, taille d'image) que du contenu de l'image (couleurs, test de pixels,…).

3.12.2.3 Filtrage d'URL :

Ceci consiste à vérifier les liens hypertextes inclus dans les messages auprès d'une base de données de « mauvais URL » préenregistrés, ou via la consultation en temps réel des listes noires disponibles sur le web. Des tentatives de masquage du lien hypertexte sont des fois utilisées par des spammeurs pour empêcher l'analyse par le filtrage d'URL.

3.12.2.4 Filtres bayésiens :

L'approche d'apprentissage automatique le plus connu dans le filtrage des spams est les classificateurs Bayes naïfs, classificateur Naïve Bayes est un classificateur probabiliste. En bref, il calcule et utilise la probabilité de certains mots/expressions apparaissant dans les exemples les plus connus (messages) afin de classer de nouveaux exemples (messages).
Naïve Bayes a été montré pour être très bien réussi à catégoriser les documents texte. Filtres bayésiens (méthode statistique) Filtres travaillé en analysant les mots du message à l'intérieur d'un e-mail pour calculer la probabilité que le message est un spam ou non. Le calcul basé sur des mots qui déterminent que le message est un spam et les mots qui déterminent que le message n'est pas du spam.

3.12.2.5 Machine à Vecteurs de Support (SVM) :

Machine à Vecteurs de Support (**SVM**) ont eu du succès dans le classement des documents texte. SVM a donné lieu à une recherche importante dans les appliquer à filtrage de spam. SVM sont des méthodes à noyaux dont l'idée centrale est d'intégrer les données représentant les documents texte dans un espace vectoriel. SVM tenter de construire une séparation linéaire entre deux classes dans cet espace vectoriel. Une machine à vecteurs de support est un classifieur linéaire binaire à marge maximale. Il peut être interprété comme trouver un hyperplan dans un espace de caractéristiques linéairement séparables qui sépare les deux classes avec une marge maximum. Les instances les plus proches de l'hyperplan sont connues comme les « vecteurs de support » car ils soutiennent l'hyperplan des deux côtés de la marge. SVM a été rapporté significative des performances sur le problème de la catégorisation de textes avec de nombreuses fonctionnalités pertinentes. SVM a également été appliquée au filtrage anti-spam. [14]

3.12.3 Quelques travaux de filtrage de spam basées sur l'apprentissage supervisé.

3.12.3.1 Drucker et al.

Drucker et al. [24] ont comparé l'efficacité du classifieur linéaire SVM avec ceux de RIPPER, Rocchio et arbres de décision. Il est la première qui a essayé un large ensemble de configurations d'expérimentations sur la sélection des termes et les différents algorithmes d'apprentissage. Ils arrivent aux conclusions suivantes :

- SVM (avec une représentation binaire) et arbres de décision (avec une représentation TF) sont les deux meilleurs classifieurs, mais les SVM permettent d'atteindre des taux de faux positifs plus bas et plus facilement.
- Dans un choix entre l'utilisation d'une liste de stopwords ou non, il est préférable qu'une liste de stopwords ne soit pas utilisée.
- L'apprentissage en utilisant les arbres de décision est énormément long. Les méthodes RIPPER et Rocchio ne sont pas performantes pour le filtrage de spam.

3.12.3.2 Saumya Goyal et al.

En 2016, Saumya Goyal et al. [25] proposent l'utilisation d'un mécanisme de détection de spam basé sur l'algorithme KNN et l'arbre de décision, ils appliquent ces algorithmes sur des ensembles de données réels de twitter. Pour analyser le mécanisme proposé l'outil WEKA7 est utilisé. Les mesures de performance telles que TP Rate, FP Rate, Precision, Recall et F-Measure sont utilisées pour évaluer le mécanisme proposé. Ils obtenaient les résultats suivants :

	TP Rate	FP Rate	Precision	Recall	F-Measure	Class
KNN	1	0.508	0.904	1	0.949	Spam
	0.492	0	1	0.492	0.659	Normal
	0.912	0.42	0.92	0.912	0.899	Weighted Avg
Arbre De Décision	1	1	0.827	1	0.905	Spam
	0	0	0	0	0	Normal
	0.827	0.827	0.683	0.827	0.748	Weighted Avg

Tableau 1 : Les mesures de performance avec l'arbre de décision [25]

Les résultats obtenus présentent que l'algorithme KNN est plus performant par rapport à l'arbre de décision.

3.12.3.3 Nurul Fitriah Rusland et al

En 2017, Nurul Fitriah Rusland et al [26] ont testé un algorithme naïf bayes pour le filtrage du spam sur deux corpus (Spam Data qui contient 9324 e-mails et 500 attributs et SPAMBASE qui contient 4601 email et 58 attributs) et tester ses performances. L'architecture du système est comme suite :

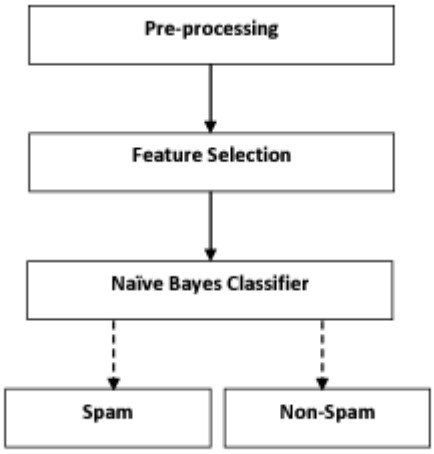
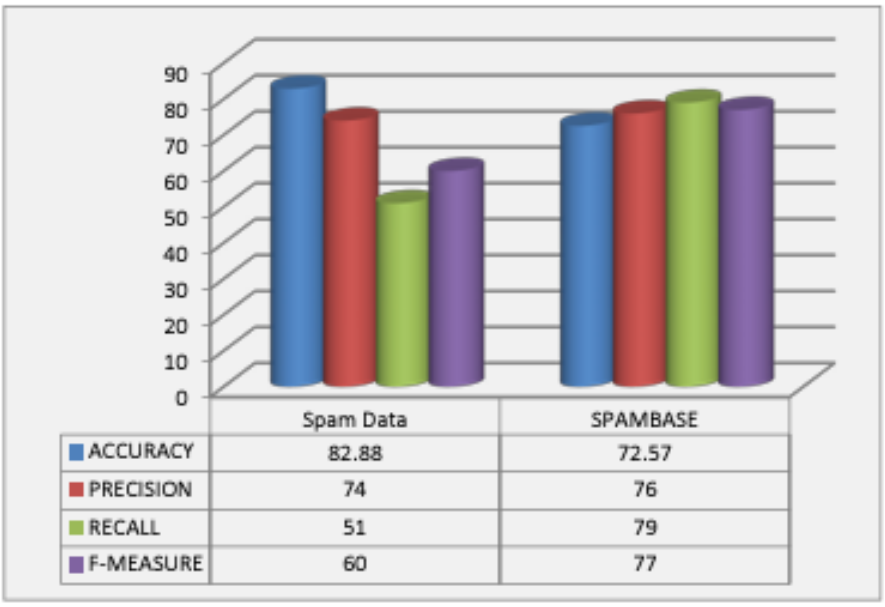


Figure 2-8 Filtre anti-spam en utilisant l'algorithme naïve bayes [26]

La performance de ce filtre est évaluée avec l'outil WEKA en fonction de leur précision, de leur rappel et de leur F-mesure. Ils obtenaient les résultats suivants :



	Spam Data	SPAMBASE
ACCURACY	82.88	72.57
PRECISION	74	76
RECALL	51	79
F-MEASURE	60	77

Figure 2-9 Résultats d'évaluation avec les deux corpus

Ils ont constaté que La performance de ce filtre est également basée sur les corpus utilisés. Comme on peut le voir, les corpus qui ont moins d'instances et d'attributs (ici SPAMBASE) peuvent donner de bons résultats.

3.12.3.4 Anju Radhakrishnan et Vaidhehi V.

Anju Radhakrishnan et Vaidhehi V. [27] utilisent deux algorithmes importants à savoir, Naïve Bayes et J48 Decision Tree et testent leur efficacité dans la classification des emails. Le corpus utilisé est Enron et la valeur TF-IDF est utilisée comme fréquence.
Les classifieurs sont également testés avec différentes tailles des attributs. Les résultats des tests sont présentés comme suite :

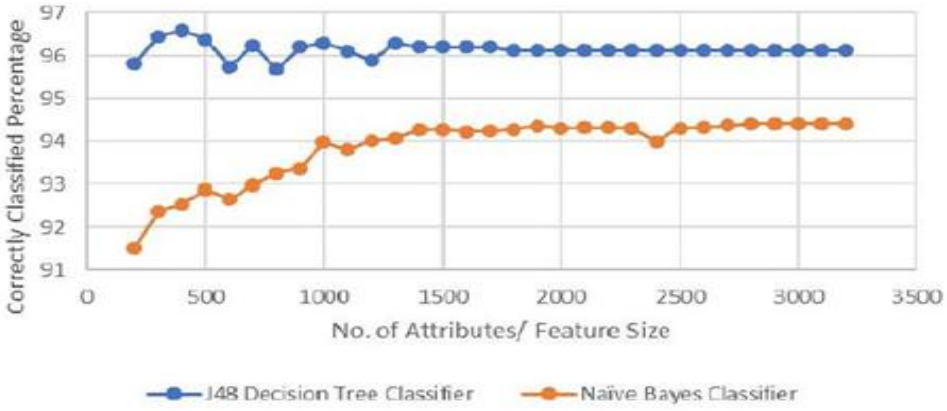


Figure 2-10 Les résultats des tests pour les deux classifieurs NB et J48 [27]

Les expériences de classification d'email ont montré que le classifieur J48 Decision Tree est plus efficace que le classifieur Naïve Bayes pour le corpus Enron. Il donne une précision de **96,5971%** dans la classification des e-mails avec une taille de 400 attributs.

3.12.3.5 Shradhanjali et Verma Toran

Shradhanjali et Verma Toran [28] proposent l'utilisation d'une nouvelle méthode pour la détection de spam en utilisant **SVM** et l'extraction des attributs qui atteint une précision de **98%**. L'architecture du système proposé est présentée dans la figure suivante :

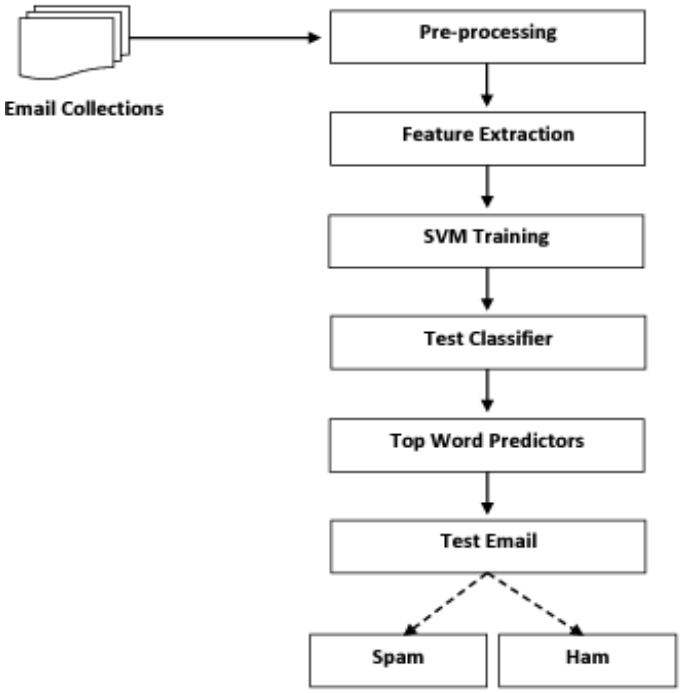


Figure 2-11 Filtre anti-spam en utilisant SVM et l'extraction des attributs [28]

- **Prétraitement** : dans l'étape de prétraitement, tous les numéros, les symboles spéciaux, les balises URL et HTML sont supprimées. Le stemming est fait pour enlever l'alphabet inutile dans les mots.
- **L'extraction des attributs** : Après le prétraitement, les attributs sont extraits.
- **Entraînement** : Après l'extraction des attributs, l'entrainement est fait. Lors de l'entrainement, les e-mails sont fournis en entrée du classifieur SVM.
- **Test de classifieur** : Après l'entrainement, les e-mails de test sont donnés pour tester l'exactitude du système. La précision est atteinte jusqu'à 98%.
- **Classification** : Enfin, le classifieur est testé avec un e-mail (La classe spam ou la classe légitime).

3.12.3.6 Jawale Diksha .S et.al

En 2018, Jawale Diksha .S et.al [29] proposent l'utilisation d'un classifieur de spam hybride **NB-SVM** qui utilise les avantages de Naïve Bayes (**NB**) et Support Vector Machine (**SVM**), **NB** est un algorithme de classification rapide et **SVM** a une grande performance en raison de leur taux de rappel et de précision élevé. Les données d'apprentissage sont d'abord traitées par l'algorithme **NB** dans lequel il calcule la probabilité pour chaque mot et message et compare avec un seuil qui classifie Les données. Les données traitées par **NB** vont à **SVM** pour améliorer la précision. L'architecture de ce classifieur est comme suite :

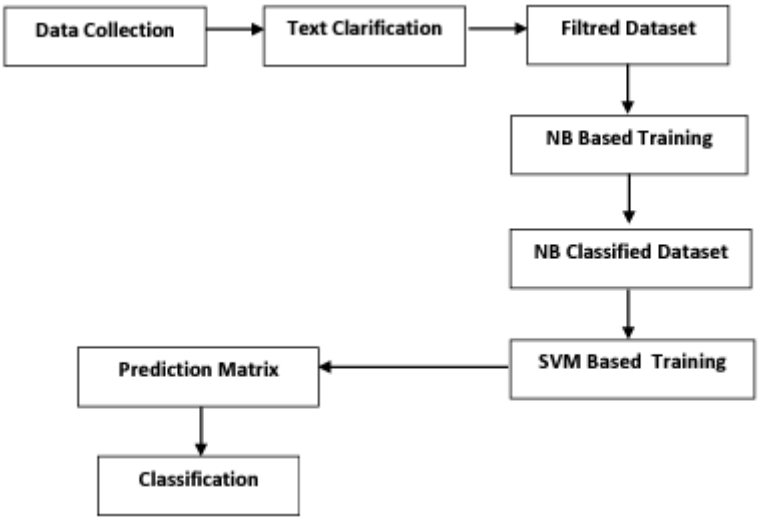


Figure 2-12 Architecture NB-SVM [29]

Avec l'utilisation de **NB**, ils obtiennent une précision de **96,65%** dans la phase d'entraînement et 95,78% dans la phase de test. Avec **SVM**, ils obtiennent une précision de **99,43%** dans la phase d'entraînement et **97,13%** dans la phase de test. La combinaison de ces deux algorithmes **NB-SVM**, donne une précision de **99,44%** dans la phase d'entraînement et **97,57%** dans la phase de test. Ce qui montre que les résultats étaient meilleurs que ceux des deux classifieurs utilisés séparément.

3.13 Conclusion

Au cours de ce chapitre nous avons présenté la définition de spam, ses objectifs et ses impacts ainsi que les diverses méthodes pour lutter contre le spam qui peuvent être classées en deux catégories : Le premier contient les solutions basées sur l'en-tête du message électronique telles que les listes noires, blanches et grises. Le deuxième groupe de solutions contient celles qui sont basées sur le contenu textuel du message telles que le filtrage basé sur l'apprentissage automatique.
Les deux approches en haut ont chacune ses faiblesses : la convergence des modèles dynamique est lente avec la taille explosive de données, alors que les modèles statistiques ont besoin de relancer constamment la phase d'apprentissage afin de s'adapter aux nouvelles données.
Nous explorons une approche basée sur la classifications bayésienne dans le chapitre suivant étant donné que c'est la technique de classification la plus répandue dans le domaine.

Chapitre III

4 Chapitre III : Technique de l'IA Pour la detection des Spams

4.1 Introduction

Au départ, le spam visait principalement des objectifs publicitaires. Aujourd'hui, il s'est considérablement développé, diversifié et complexifié, pour atteindre de plus en plus souvent des objectifs malveillants. En effet Le spam s'est non seulement développé en termes de volume, mais également en termes de contenu.

De nombreuses méthodes ont été développées pour lutter contre les spam. La situation évolue pourtant de façon positive avec l'apparition d'une nouvelle catégorie de méthodes, le filtrage par approche probabiliste.

Les modèles probabilistes sont très efficaces pour la détection des spam. Dans ce chapitre, nous présentons une de méthode probabiliste qui est la méthode bayésienne en proposant l'utilisation de deux lois de probabilités, à savoir la loi normale, et la loi de Rice, ainsi que la combinaison des résultats des deux lois, en fournissant ainsi un filtre collaboratif.

4.2 Principe de l'approche

Quel que soit les caractéristiques qu'on peut extraire des e-mails, le filtrage de ces dernier demeure incertain et imprécis, étant donné la grande variabilité des textes formant les entêtes et les corps des e-mails. Dans ce travail, nous allons plutôt chercher à faire coopérer deux filtres différents pour renforcer la certitude de décision de chacun d'eux.
Pour ce faire, nous avons considéré deux caractéristiques pour les textes des e-mails:

1) Le nombre de mots total composant un e-mail .
2) Le nombre de mots suspects dans le texte et le titre du mail.

Le choix des caractéristiques est un sujet à part entière, et il fait l'objet de travaux de recherche indépendants. Cependant l'approche que nous proposons dans ce travail reste valable quel que soit les caractéristiques extraites des emails, à condition qu'on peut définir une loi de probabilité à densité pour chacune d'elle.

Afin de tester la collaboration des filtres anti-spam, nous avons considéré deux filtres distincts: **un Filtre Normal**, et **un Filtre Rice**.

Dans les deux cas de filtres nous classifions les e-mails selon le maximum à postériori (**MAP**) de la loi de bayes montre un diagramme en blocs de l'approche proposée. Nous détaillons dans la suite des sections de ce chapitre, tous les éléments de ce diagramme.

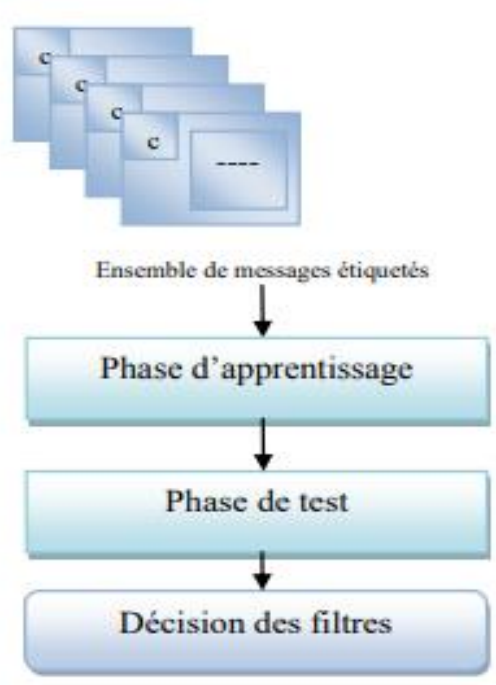


Figure 3-1 : Diagramme général de notre approche

Nous détaillons les deux phases, d'apprentissage et de test, comme suit :

a) ***Phase d'apprentissage***

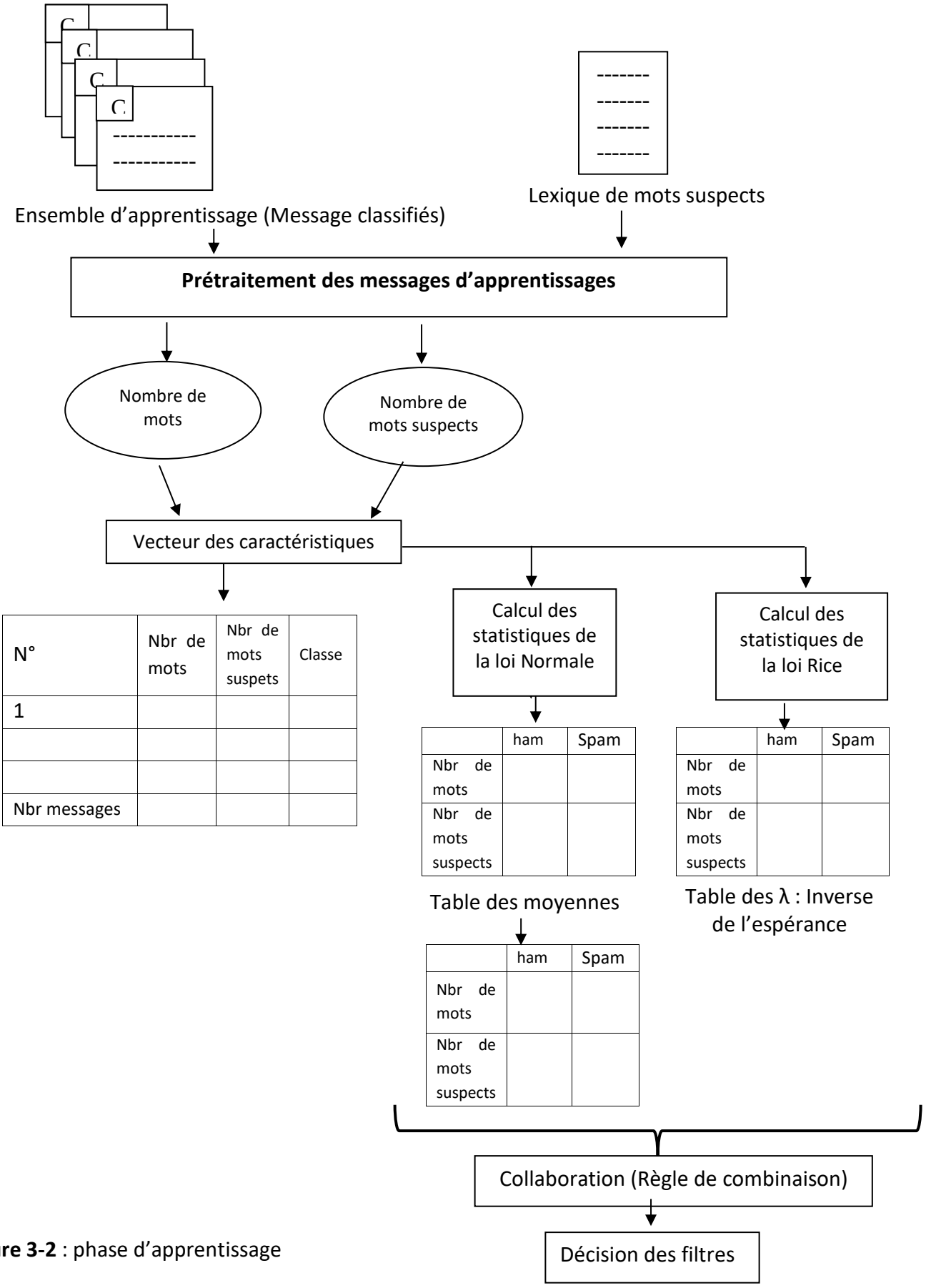


Figure 3-2 : phase d'apprentissage

b) ***Phase de teste***

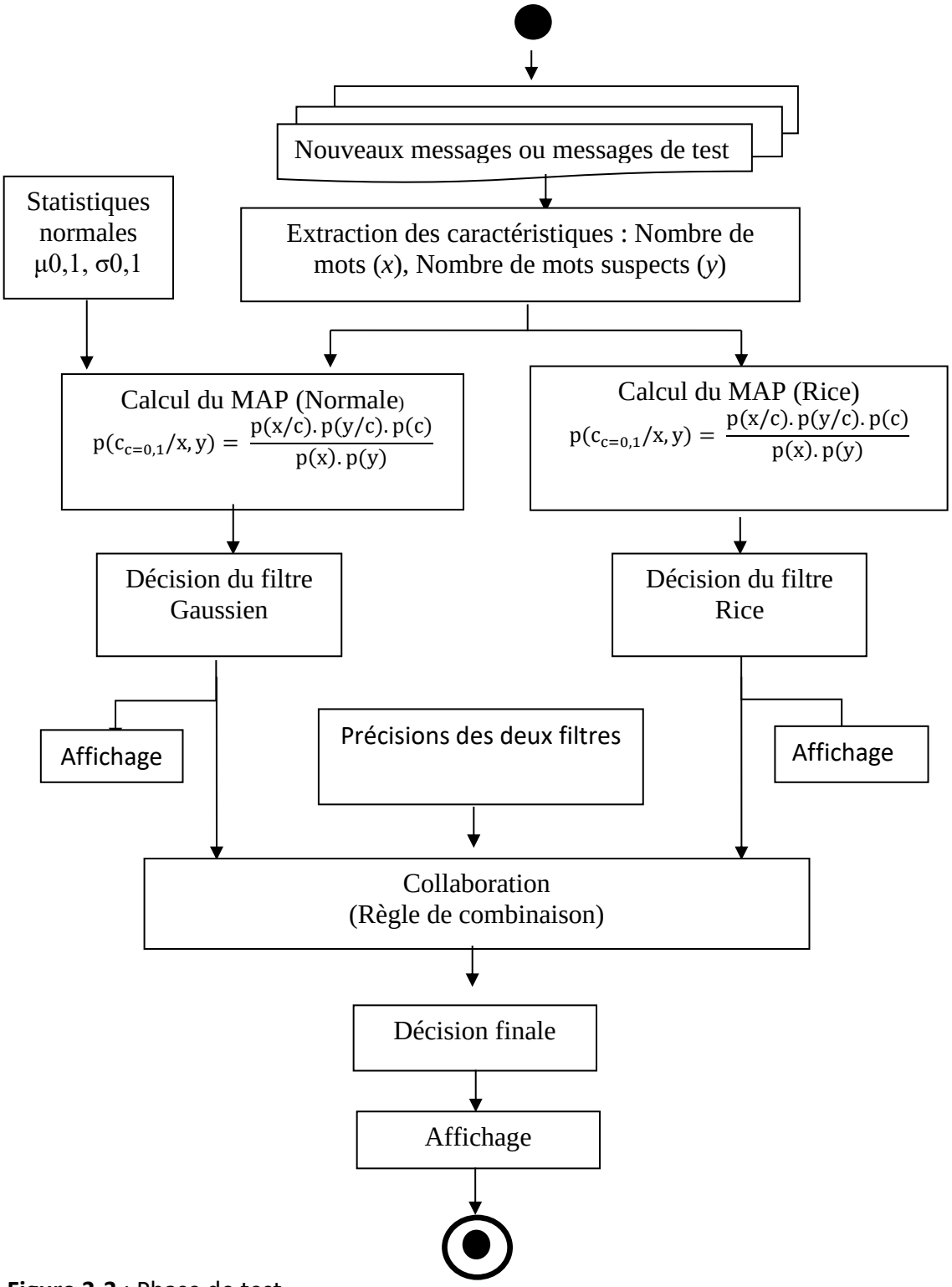


Figure 3-3 : Phase de test

4.2.1 Le Filtre Bayésien

Tous d'abord, nous rappelons que la théorie de la décision bayésienne est basée sur l'hypothèse que le problème de décision est posé dans un contexte probabiliste et que toutes les probabilités pertinentes sont connues. En prenant deux événements A et B et en ayant P(A) ≠ 0.
Elle est exprimée par :

$$p(A|B)=\frac{p(A|B).p(B)}{p(A)}$$

avec :

- P(A) la probabilité a priori (ou "inconditionnelle" ou "marginale") de A, dans le sens qu'elle ne prend en compte aucune information concernant B.
- P(B|A) est la probabilité conditionnelle de B sachant A. Elle est aussi appelée la probabilité a posteriori puisqu'elle dépend de l'événement B.
- P(A|B) est la probabilité conditionnelle de A sachant B. Elle est aussi appelée la vraisemblance.
- P(B) est la probabilité a priori ou marginale de B

4.2.2 Filtrage Bayésien des spams

Le filtrage bayésien des spam, en référence au théorème de Bayes, est une technique statistique de détection de pourriels s'appuyant sur la classification naïve bayésienne.
Les filtres bayésiens fonctionnent en établissant une corrélation entre la présence de certains éléments (en général des mots, parfois d'autres choses) dans un message et le fait qu'ils apparaissent, en général, dans des messages indésirables ou dans des messages légitimes (ham).
Cette technique est puissante dans le traitement des courriers électroniques indésirables. Elle s'adapte aux habitudes de courrier des uns et des autres et produit un taux de faux positifs suffisamment bas pour être acceptable.
Dans le cadre de ce travail, nous allons introduire, dans le théorème de Bayes, la notion de classe Ci et de message mi. la classe Ci peut être spam ou ham. Ainsi, l'équation devient :

$$p(Ci|m_i)=\frac{p(m_i|Ci).p(Ci)}{p(m_i)}$$

$p(Ci|m_i)$ est la probabilité qu'un message m_i appartienne à la classe *Ci.*
p(Ci) est la probabilité à priori qui est estimée par le pourcentage d'exemples appartenant à la classe Ci dans le corpus d'apprentissage.
Le dénominateur *p( ),* étant invariant, et ne sera pas donc pris en considération dans calcul et la décision.

p $(m_i$ */Ci)* est la probabilité conditionnelle de vraisemblance. Elle est liée aux paramètres de classification des messages, à savoir la moyenne et la variance des différentes variables.

Cette probabilité sera déterminée par deux lois différentes : la loi de Gausse et la loi uniforme. La première admit deux paramètres pour chaque variable (caractéristique), à savoir la moyenne, et l'écart-type. Par contre la deuxième, elle admit un seul paramètre pour chaque variable, et qui est l'inverse de son espérance mathématique .

4.3 Corpus de messages

Nous nous situons dans une approche supervisée, où il est nécessaire de disposer de données d'apprentissage préalablement étiquetées. Dans le domaine de la sécurité de la messagerie, on fait recours aux corpus de messages, préparés pour l'entrainement des filtres anti-spam.

Un corpus de message est donc un ensemble de messages examiné par un ou des experts dont l'objectif est de les qualifier en messages normaux (ham) ou messages spam.

L'analyse textuelle d'un corpus de message est, classiquement, basée sur la fréquence de la sélection de mots. L'objectif général est de pouvoir discriminer les courriels pertinents ou encore de définir un modèle personnalisé de leur détection. Il s'agit donc d'un modèle susceptible de prévoir la qualité d'un message reçu en fonction de son contenu.

Les filtres de contenu examinent le contenu du message pour déterminer s'il s'agit d'un spam ou d'un ham (légitime). Ces filtres de contenu essayent de lire le texte afin d'examiner son contenu par mots-clés.

Dans ce travail, l'examen du contenu est basé sur le nombre total de mots du message (variable *x*) et sur le nombre d'occurrences des mots suspects (variable *y*).

Les différents mots forment une base pour la création de vecteurs de données numériques correspondant à une collection de messages.

Chaque vecteur correspond à un message et chaque ligne du vecteur représente un message. Ainsi, une cellule du vecteur est une mesure d'une caractéristique (correspondant à la colonne) pour un message (correspondant à une ligne).

Puisque la prédiction de la nature du message (spam ou ham) est notre objectif, nous avons inséré une colonne de plus (une classe) contenant la réponse correcte dans chaque vecteur message. Lors de la préparation des données pour une méthode d'apprentissage, la réponse correcte sera disponible à partir des étiquettes de message (0 pour ham et 1 pour spam).

Dans la collection de messages dont on connait la classe, les mots suspects répétés fréquemment sont comptés et rassemblés dans un dictionnaire.

N° Message	Nombre de mots	Nombre de mots suspects	Classe
Message1			
Message2			
:			
:			
Message N			

tableau 2 : Représentation d'un corpus de messages

4.4 Principe de détection

On nous donne un vecteur de messages d'entrée M = [m1, m2, m3 .., mn] où chaque pourrait être classé dans l'une des k classes C1, C2... Ck. Pour réaliser cette classification, nous utilisons l'algorithme Naïve Bayes pour modéliser les probabilités **P (Ci | M)** pour chacune des classes C1, C2... Ck. Ensuite, nous étiquetons la classe de M en tant que Ci telle que P (Ci | M) soit la plus élevée. Pour calculer P (Ci | M), nous utilisons le théorème de Bayes:

Dans le problème de la détection des spam, nous utilisons deux classes seulement : C1=ham et C2= spam.

Les probabilités **P1** et **P2** pour qu'un message **mi** soit **spam** ou **ham** respectivement, prendront la forme suivante :

$$P_1(\text{ham}|m_i)=\frac{p(m_i|\text{ham}).p(\text{ham})}{p(m_i)}$$

$$P2(\text{spam}|m_i)=\frac{p(m_i|\text{spam}).p(\text{spam})}{p(m_i)}$$

Si p1>p2 alors mi est un **ham**. Sinon, mi est un **spam**.

p(ham) est la probabilité dans l'absolu qu'un message quelconque soit un ham. Elle est donnée par:

$$P(\text{ham})=\frac{\text{nombre de ham}}{\text{nombre de message}}$$

p(spam) est la probabilité dans l'absolu qu'un message quelconque soit un spam. Elle est donnée par:

$$P(\text{spam})=\frac{\text{nombre de spam}}{\text{nombre de message}}$$

Dans ce travail, les probabilités conditionnelles **$p(m_i|spam)$** et **$p(m_i|ham)$** seront déterminées par deux méthodes : Selon la loi Gaussienne et selon la loi Rice.

4.4.1 Méthode Gaussienne

La densité de probabilité de Gauss, pour une variable *x* s'écrit :

$$f(x)=\frac{1}{\sigma\sqrt{2\pi}}e^{\frac{-1/2(X-\mu)^2}{\sigma^2}}$$

où μ et σ sont respectivement la moyenne et l'écart type de *x.*

Probabilité qu'un message soit un spam

Pour calculer la probabilité qu'un message mi soit un spam, on suppose que chaque message du corpus suit une loi normale Gaussienne :

$$p(mi|spam)=\frac{1}{2\pi\sigma_x^s\sigma_y^s}e^{-\frac{1}{2}\left[\frac{(x-\mu_x^s)^2}{\sigma_x^{s2}}+\frac{(y-\mu_y^s)^2}{\sigma_y^{s2}}\right]}$$

tel que :

-La moyenne et l'écart type :

$$\delta_x^s=\sqrt{\overline{x_x^{s2}}-\overline{x_x^{s2}}}$$

$$\delta_y^s=\sqrt{\overline{y_y^{s2}}-\overline{y_y^{s2}}}$$

Où :

$$\overline{x_x^{s2}}=\frac{1}{N}\sum(x_x^s)_1^2$$

$$\overline{x_x^{s2}}^2=\left(\frac{1}{N}\sum(x_x^s)_1^2\right)^2$$

$$\overline{y_y^{s2}}=\frac{1}{N}\sum(y_y^s)_1^2$$

$$\overline{y_y^{s2}}^2=\left(\frac{1}{N}\sum(y_y^s)_1^2\right)^2$$

Où ***N*** est le nombre total de spam dans le corpus.

$x_x^s i$: est le nombre total de mots dans un message spam.

$y_y^s i$: est le nombre de mots suspects dans un message spam.

$$\mu_x^s=\frac{\sum tailles\ des\ messages\ spam}{Nonbre\ de\ spam}$$

$$\mu_y^s=\frac{nombre\ de\ mot\ suspect\ dans\ les\ messages\ spams}{nombre\ de\ spam}$$

Probabilité qu'un message soit un ham

De même pour la probabilité qu'un message m_i soit un ham :

$$p(m_i|ham) = \frac{1}{2\pi\sigma_x^h\sigma_y^h} e^{-\frac{1}{2}[\frac{(x-\mu_x^h)^2}{\sigma_x^{h\,2}} + \frac{(y-\mu_y^h)^2}{\sigma_y^{h\,2}}]}$$

tel que :
La moyenne et l'écart type :

$$\delta_x^h = \sqrt{\overline{x_x^{h\,2}} - \overline{x_x^h}^2}$$

$$\delta_y^h = \sqrt{\overline{y_y^{h\,2}} - \overline{y_y^h}^2}$$

Où

$$\overline{x_x^{h\,2}} = \frac{1}{N}\sum (x_x^h)_1^2$$

$$\overline{x_x^h}^2 = \left(\frac{1}{N}\sum (x_x^h)_1^2\right)^2$$

$$\overline{y_y^{h\,2}} = \frac{1}{N}\sum (y_y^h)_1^2$$

$$\overline{y_y^h}^2 = \left(\frac{1}{N}\sum (y_y^h)_1^2\right)^2$$

Où ***N*** est le nombre total de ham dans le corpus.

$x_x^s i$: est le nombre total de mots dans un message spam.
$y_y^s i$: est le nombre de mots suspects dans un message spam.

$$\mu_x^h = \frac{\sum tailles\ des\ messages\ ham}{Nonbre\ de\ ham}$$

$$\mu_y^h = \frac{nombre\ de\ mot\ suspect\ dans\ les\ messages\ ham}{nombre\ de\ ham}$$

4.4.2 Méthode Rice

En statistiques et théorie des probabilités, la loi de Rice, nommée d'après Stephen O. Rice (en) (1907–1986), est une loi de probabilité à densité (c'est-à-dire continue).

C'est une généralisation de la loi de Rayleigh utilisée pour décrire le comportement d'un signal radio qui se propage selon plusieurs chemins (multipath) avant d'être reçu par une antenne.

Soient deux variables de Gauss centrées, indépendantes, de même variance σ2. Si on considère qu'elles représentent les deux coordonnées d'un point d'un plan, la distance de ce point à l'origine suit une loi de Rayleigh :

$$f(x,\sigma) = \frac{x}{\sigma^2}\exp\left(\frac{-x^2}{2\sigma^2}\right).$$

En supposant que la distribution est centrée sur un point de coordonnées (ν cos θ, ν sin θ) (coordonnées polaires (ν , θ)), la densité de probabilité devient :

$$f(x|\nu,\sigma) = \frac{x}{\sigma^2}\exp\left(\frac{-(x^2+\nu^2)}{2\sigma^2}\right) I_0\left(\frac{x\nu}{\sigma^2}\right)$$

où $I_0(z)$ est la fonction de Bessel modifiée de première espèce et d'ordre 0.

Les premiers moments (non centrés) sont :

$$\mu_1 = \sigma\sqrt{\pi/2}\; L_{1/2}(-\nu^2/2\sigma^2)$$
$$\mu_2 = 2\sigma^2 + \nu^2$$
$$\mu_3 = 3\sigma^3\sqrt{\pi/2}\; L_{3/2}(-\nu^2/2\sigma^2)$$
$$\mu_4 = 8\sigma^4 + 8\sigma^2\nu^2 + \nu^4$$
$$\mu_5 = 15\sigma^5\sqrt{\pi/2}\; L_{5/2}(-\nu^2/2\sigma^2)$$
$$\mu_6 = 48\sigma^6 + 72\sigma^4\nu^2 + 18\sigma^2\nu^4 + \nu^6$$
$$L_\nu(x) = L_\nu^0(x) = M(-\nu,1,x) = {}_1F_1(-\nu;1;x)$$

où, $L_\nu(x)$ représente un polynôme de Laguerre et *M* désigne la fonction hypergéométrique confluente.

Pour le cas $\nu = 1/2$:

$$L_{1/2}(x) = {}_1F_1\left(-\frac{1}{2};1;x\right)$$
$$= \mathrm{e}^{x/2}\left[(1-x)\,I_0\left(\frac{-x}{2}\right) - xI_1\left(\frac{-x}{2}\right)\right].$$

Généralement les moments sont donnés par

$$\mu_k = s^k 2^{k/2}\,\Gamma(1+k/2)\,L_{k/2}(-\nu^2/2\sigma^2),$$

où $s = \sigma^{1/2}$.

Lorsque *k* est pair, les moments deviennent des polynômes en *σ* et *ν*.

Distributions liées

La variable $r = \sqrt{x^2 + y^2}$ suit une loi de Rice R~Rice(σ , v) à condition que
X ~N *(vcosθ, σ^2)* et Y~N *(vsinθ, σ^2)* soient deux variables gaussiennes **indépendantes**.

Pour obtenir une variable R~Rice(σ , v) , on peut considérer une autre procédure :
Tirer *P* selon une loi de Poisson, de paramètre $\lambda = \frac{\nu^2}{2\sigma^2}$
Tirer *X* selon une loi du χ^2 avec 2*P* + 2 degrés de liberté.
Poser $R = \sigma\sqrt{x}$.
Si $R \sim \mathrm{Rice}\,(\sigma, \nu)$ alors R^2 suit une loi du χ^2 non centrée, à 2 degrés de liberté et un paramètre de non-centralité v^2.

Probabilité qu'un message soit un spam

Pour calculer la probabilité qu'un message mi soit un spam, on suppose que chaque message du corpus suit une loi de Rise :

$$P(m_i/spam)=\frac{x^s * y^s}{\delta_x^{s^2} * \delta_y^{s^2}} \exp\left(-\frac{x^{s^2}}{\delta_x^{s2}} - \frac{y^{s^2}}{\delta_y^{s2}}\right)$$

tel que : $\delta_x{}^s$, $\delta_y{}^s$ sont les écart-types des variables x^s et y^s.

Probabilité qu'un message soit un ham

De même pour la probabilité qu'un message m_i soit un ham :

$$P(m_i/ham)=\frac{x^h * y^h}{\delta_x^{h^2} * \delta_y^{h^2}} \exp\left(-\frac{x^{h^2}}{\delta_x^{h2}} - \frac{y^{h^2}}{\delta_y^{h2}}\right)$$

tel que : $\delta_x{}^h$, $\delta_y{}^h$ sont les écart-types des variables x^h et y^h.

Collaboration des deux filtres

Pour un courriel à l'entrée du système, ou un courriel de test, nous appliquons les deux filtres et nous obtenons les probabilités suivantes :

$\boldsymbol{p_g^h}$ **:** Probabilité selon le filtre Gaussien que le courriel est «ham».
$\boldsymbol{p_g^s}$ **:** Probabilité selon le filtre Gaussien que le courriel est «spam».
$\boldsymbol{p_r^h}$ **:** Probabilité selon le filtre Rice que le courriel est «ham».
$\boldsymbol{p_r^s}$ **:** Probabilité selon le filtre Rice que le courriel est «spam».

Sur la base de ces probabilités, nous exprimons les règles de décision finales :

- Si (p_g^h<p_g^s) et (p_r^h < p_r^s) alors «spam» .
- Si (p_g^h > p_g^s) et (p_r^h > p_r^s) alors «Ham» .
- Si (p_g^h > p_g^s) et (p_r^h < p_r^s) alors : si(p_g^h × précision$_g$> p_r^s× précision$_r$) alors «ham» Sinon «spam».
- Si (p_g^h <p_g^s) et (p_r^h > p_r^s) alors : si(p_g^h × précision$_g$>p_r^s × précisior$_r$) alors «ham» Sinon «spam».
- Si (p_g^h >p_g^s) et (p_r^h < p_r^h) alors : si(p_g^h × précision$_r$>p_g^s × précision$_g$) alors «spam» Sinon «ham».

Avec précision**g**, précision**r** sont respectivement les précisions des filtres Gaussien et Rice obtenues d'une manière générale pour une loi *l* comme suit :

$$\text{Précision}_l = \frac{nombre\ de\ message\ correctement\ classés\ selon\ la\ \text{loi}\ l}{nombre\ de\ message}$$

Nous représenterons dans la suivante la structure fonctionnelle de notre application en utilisant les diagrammes de cas d'utilisations, et nous présenterons aussi une vue dynamique par la modélisation des diagrammes de séquences selon la notation du langage UML.

4.5 Modélisation par un langage de conception (UML)

Le langage UML (Unified Modeling Language, ou langage de modélisation unifié) a été pensé pour être un langage de modélisation visuelle commun, et riche sémantiquement et syntaxiquement. Il est destiné à l'architecture, la conception et la mise en œuvre de systèmes logiciels complexes par leur structure aussi bien que leur comportement. L'UML a des applications qui vont au-delà du développement logiciel, notamment pour les flux de processus dans l'industrie.

Il ressemble aux plans utilisés dans d'autres domaines et se compose de différents types de diagrammes. Dans l'ensemble, les diagrammes UML décrivent la limite, la structure et le comportement du système et des objets qui s'y trouvent.

L'UML n'est pas un langage de programmation, mais il existe des outils qui peuvent être utilisés pour générer du code en plusieurs langages à partir de diagrammes UML. L'UML a une relation directe avec l'analyse et la conception orientées objet. [19]

4.5.1 Diagrammes de cas d'utilisation

En langage de modélisation unifié (UML), un diagramme de cas d'utilisation peut servir à résumer les informations des utilisateurs de votre système (également appelés acteurs) et leurs interactions avec ce dernier. La création de ce type de diagramme requiert un ensemble de symboles et de connecteurs spécifiques. Lorsqu'ils sont bien conçus, les diagrammes de cas d'utilisation peuvent aider votre équipe à collaborer et représenter :
Les scénarios dans lesquels votre système ou application interagit avec des personnes, des organisations ou des systèmes externes ;
Les objectifs que votre système ou application permet aux entités (appelées acteurs) d'atteindre ;
La portée de votre système. [19]

Le diagramme de cas d'utilisation de notre étude est le suivant :

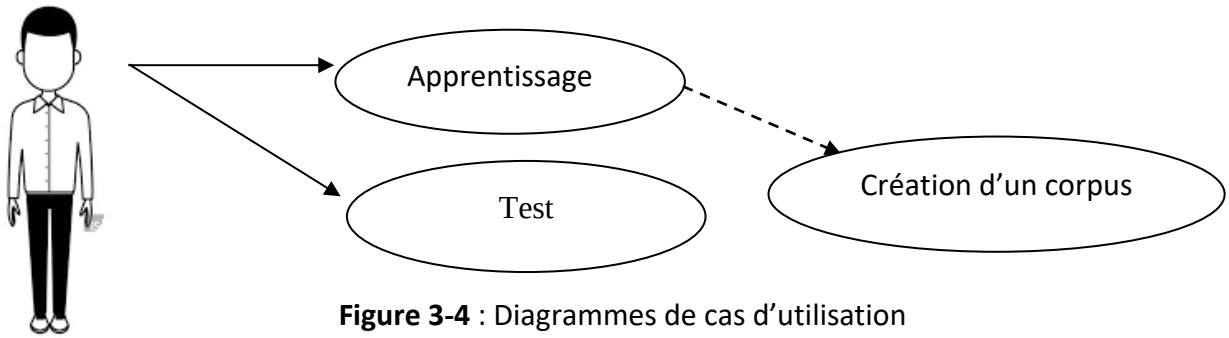


Figure 3-4 : Diagrammes de cas d'utilisation

4.5.2 Diagramme d'activité

Le diagramme d'activité est un diagramme comportemental d'UML, permettant de représenter le déclenchement d'événements en fonction des états du système et de modéliser des comportements parallélisables (multi-threads ou multi-processus). [19]
Le diagramme d'activité est également utilisé pour décrire un flux de travail. Dans ce travail l'algorithme bayésien classifie les données en deux étapes :

- **Etape d'apprentissage** : en utilisant un échantillon d'apprentissage, l'algorithme estime les paramètres de la distribution de probabilité, en supposant que les messages soient conditionnellement indépendants étant donné la classe.

- **Etape de test :** pour chacun des messages de test, l'algorithme calcule la probabilité a posteriori que ce message appartient à chacune des classes. L'algorithme classifie alors cette donnée selon la plus grande probabilité a posteriori.

Les diagrammes d'activités des différentes étapes du système sont présentés ci-après.

a) Diagramme d'activité d'apprentissage

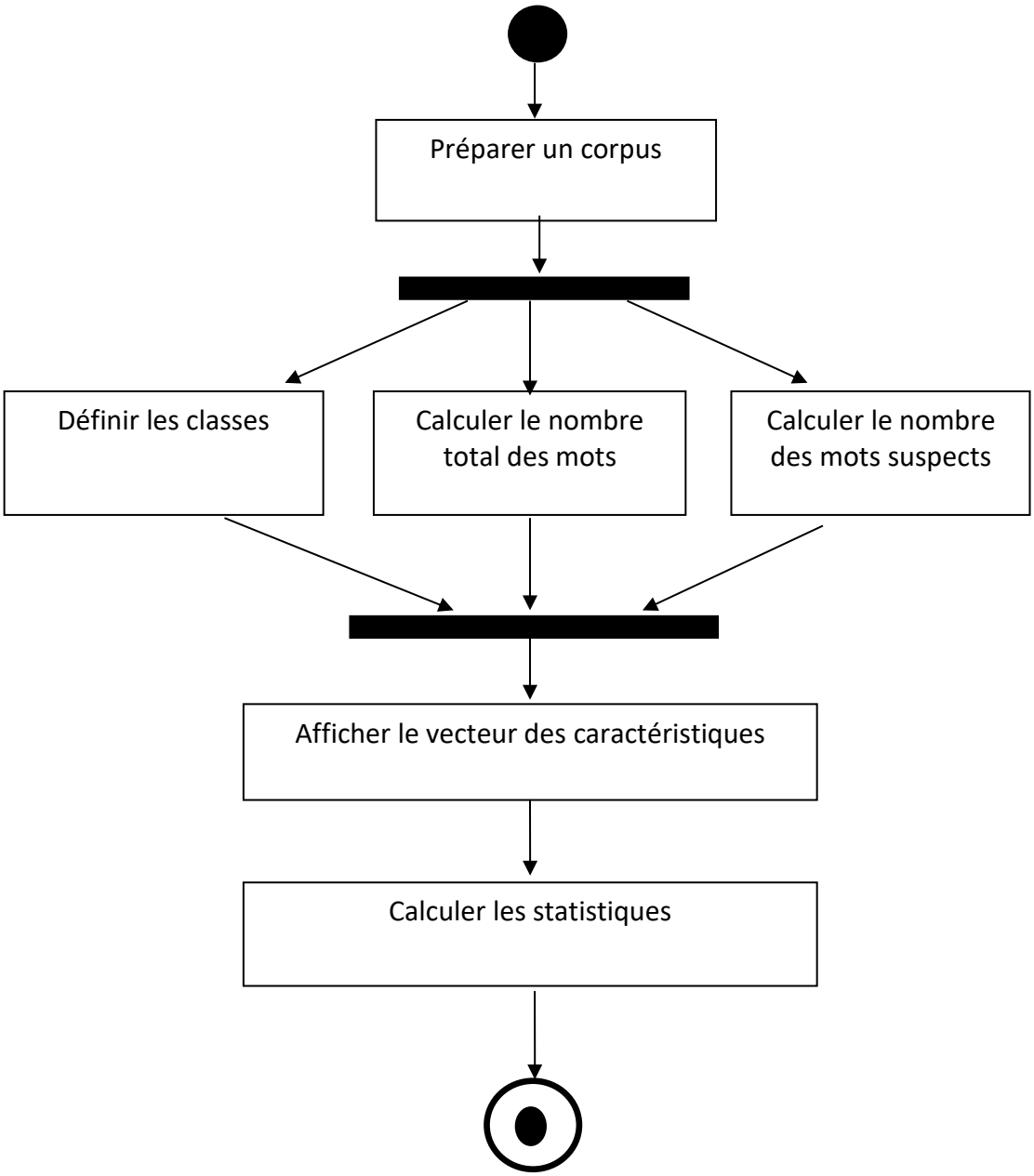


Figure 3-5 : Diagramme d'activite d'apprentissage

b) Diagramme d'activité de test

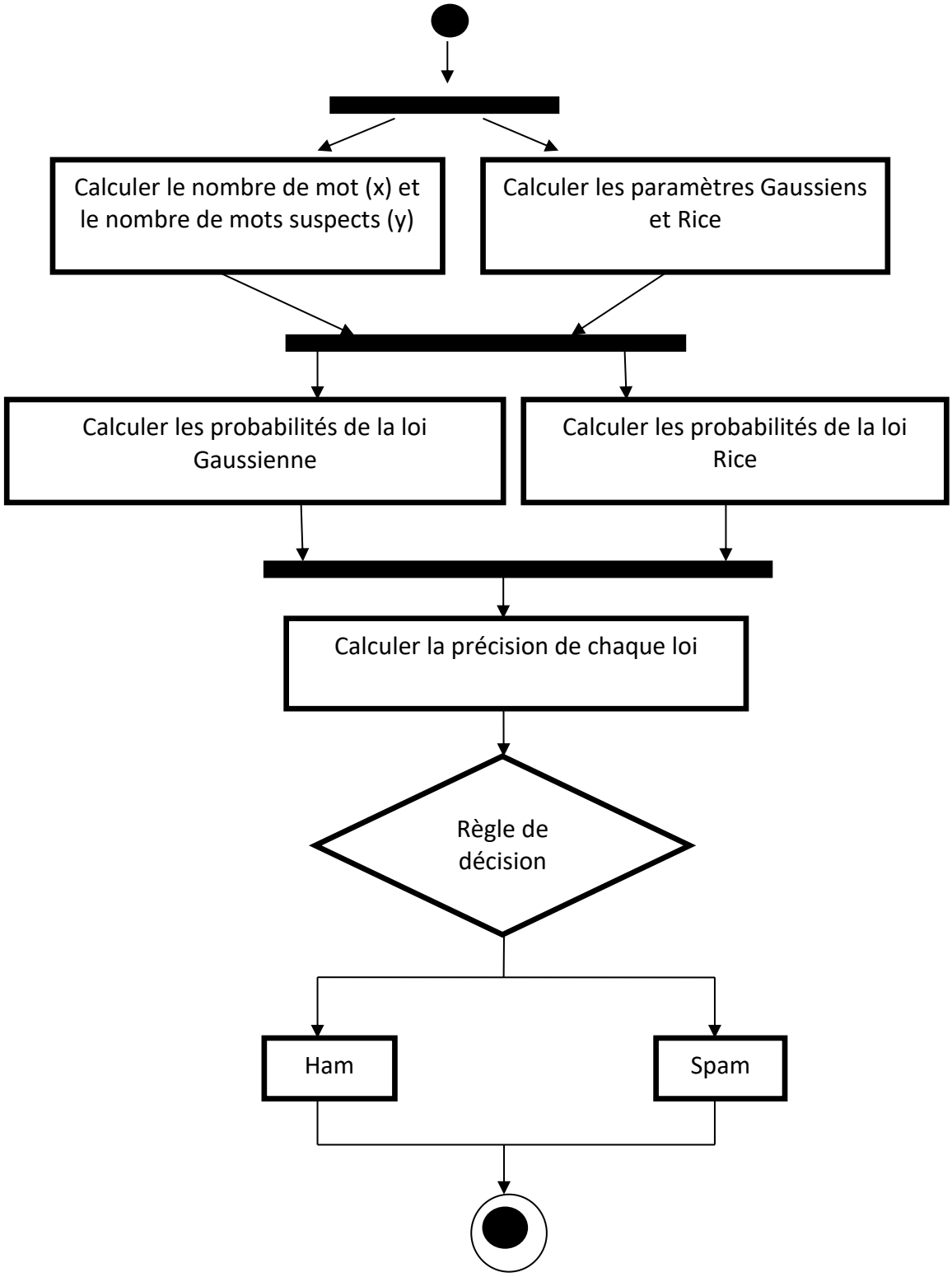


Figure 3-6 : Diagramme d'activite de test

4.6 Architecture du système

L'architecture de notre modèle de filtrage anti-spam est représentée dans la figure Tout d'abord, en utilisant le corpus SMS Spam Collection (SMS Spam Collection Dataset Kaggle).
Le corpus de messagerie va passer par la phase de préparation ou prétraitement puis en utilisant les deux algorithmes d'apprentissage (Gaussien, Rice) pour construire un modèle qui permet de classifier les nouveaux messages.

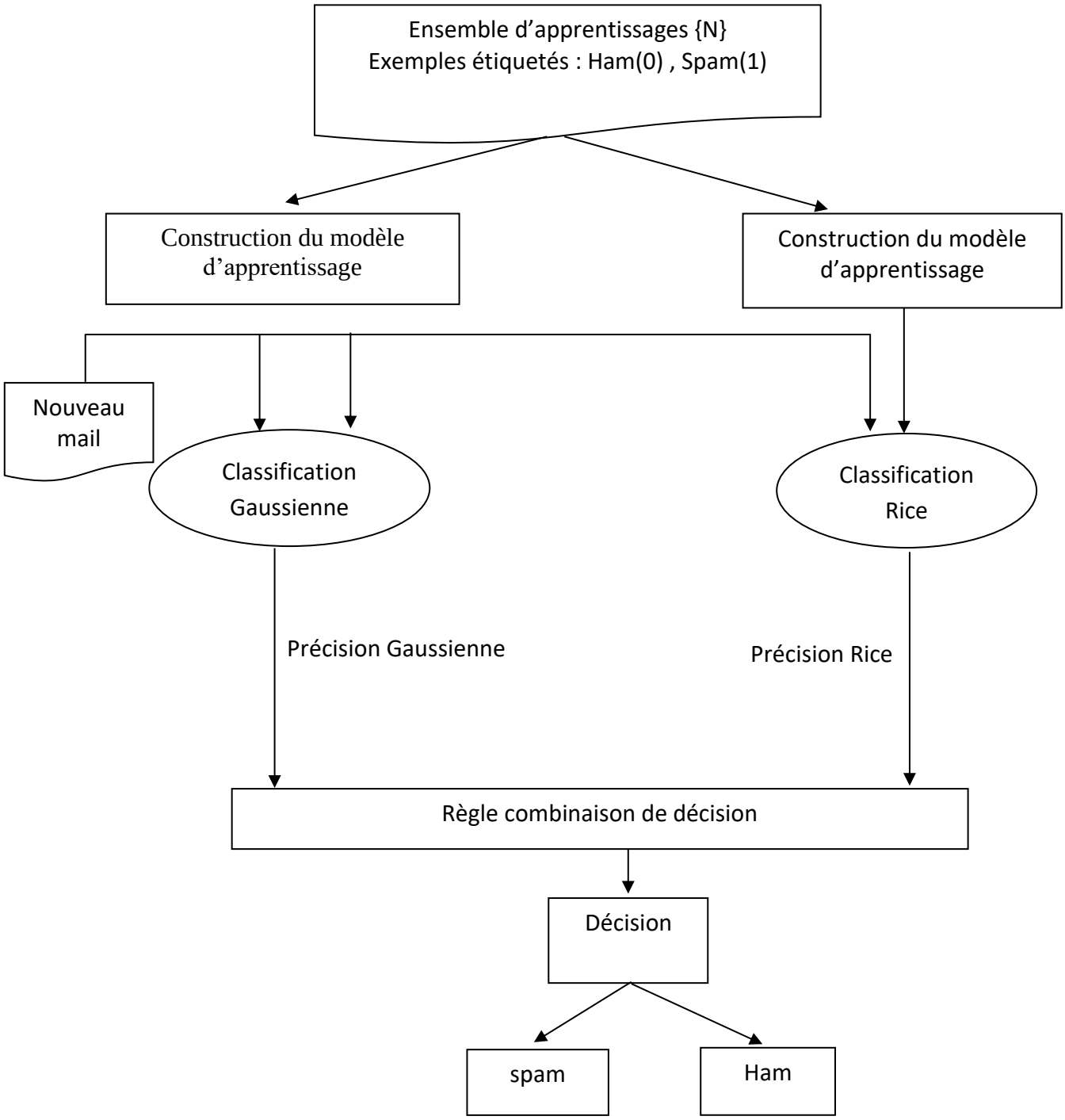


Figure 3-7 : Architecture du système

4.7 Conclusion

Dans ce chapitre nous avons détaillé notre méthode de filtrage collaboratif dont l'objectif de cette dernière est d'améliorer significativement la lutte contre le Spam. La méthode que nous avons proposée consiste à combiner deux filtres, un fonctionnant selon la loi de probabilité Gaussienne et l'autre selon la loi de probabilité Rice.

La décision finale est prise selon un ensemble de règle de décision stochastique, et prenant en compte la précision des deux filtres utilisés. Pour bien illustrer notre modèle, nous l'avons exprimé en utilisant les diagrammes appropriés du langage UML.

Au chapitre suivant, nous détaillerons la mise en œuvre du système proposé, et nous présentons quelques résultats de test, obtenu sur un corpus réel de courriels.

Chapitre IV

5 Chapitre IV : Implementation & Test

5.1 Introduction

Dans ce chapitre, nous présentons l'essentiel de l'implémentation de notre système, en commençant pas le langage de programmation et l'environnement de développement, à savoir **Delphi.** Nous consacrons également une partie à la description de l'application via ses interfaces, permettant à son utilisateur de bien tirer profit de ses fonctionnalités. A la fin du chapitre, nous présentons un test et nous commentons les résultats obtenus.

5.2 Delphi

Créé en 1995 par l'éditeur de logiciels Borland Software Corporation, **le Delphi** est d'abord un langage de programmation de haut niveau orienté objet, mais aussi un environnement de développement intégré (EDI) fonctionnant sur Windows.
Une version spéciale pour Linux a vu le jour en 2001, toujours créée par Borland. C'est une interface qui aide les programmeurs dans leur développement de logiciels exécutables. Généralement, un EDI comporte un éditeur de texte, un compilateur, un débogueur, un éditeur de liens.

Les utilisations du Delphi

Désormais, il existe des versions du Delphi pour presque tous les supports d'exploitation : Windows et Linux, mais aussi MacOS, iOS et Android. Le Delphi correspond aussi à un langage de programmation orienté objet hérité du langage Pascal.

Cet environnement est particulièrement adapté pour :

- La programmation d'applications graphiques pour le mobile
- La gestion d'une base de données
- Le développement de logiciels d'entreprise.

Les avantages du langage Delphi

Ce qui a rendu le Delphi de Borland populaire est qu'il bénéficie d'une interface :

- Intuitive, pour développer des applications graphiques facilement.
- Rapide, puisqu'il n'y a qu'une seule phase de compilation.
- Facile à utiliser, car construit sur l'interaction entre un éditeur d'interface graphique et un éditeur de code source.
- Flexible, et donc adaptée pour mener un projet dans son ensemble.
- Pédagogique. Il a été souvent servi à enseigner les bonnes pratiques de base de la programmation d'applications aux ingénieurs débutants.

Les inconvénients du Delphi

Le langage Delphi n'est pas open source, c'est un outil appartenant à la société Embarcadero. En conséquence, son destin est entre les mains d'une compagnie, et il n'y a pas de comité indépendant capable d'harmoniser son usage. Résultat, certaines fonctionnalités sont introduites avec un temps de retard, ce qui est frustrant pour la communauté qui l'utilise. Par ailleurs, pour beaucoup de développeurs, il est trop proche du Pascal. Pour ceux qui sont habitués à programmer en Java ou en C++, ce n'est pas très attractif.

Ce langage a su se démarquer des autres langages comme C ou C++ par son effort de simplicité et de gain de productivité pour le développeur. Avec des temps de compilation records et une absence de séparation entre l'implémentation et l'interface, le Delphi a même réussi à concurrencer Microsoft avec son langage Visual Basic. Dans les années 90, Microsoft ne tarde pas à embaucher la majeure partie de l'équipe ayant conçu Delphi afin d'intégrer le système dans le fonctionnement de Visual Basic [20].

5.3 Le pascal objet

Delphi implémente une version *orientée objet* du *langage Pascal* : le *Pascal Objet*, renommé Langage de programmation Delphi au fil des modifications apportées par Borland. Le Pascal Objet de Delphi possède plusieurs avantages : *typage fort*, contrôle strict du *compilateur* pour éviter les erreurs de mémoire, de débordement, gestion intégrée des *chaînes de caractères* et des tableaux dynamiques, etc. La compilation ne se fait qu'en une seule passe et il n'y a pas de séparation entre l'implémentation et l'interface comme en *C* ou en C++ : la génération d'un projet Delphi est donc très rapide, ce qui a accru à sa sortie la popularité de l'outil vite réputé pour ses temps de compilation record. De plus, sa parenté avec le Pascal rebute de nombreux programmeurs plus habitués à des styles de programmation proches de *Java* ou C. Enfin, il n'est ni standardisé ni géré par un comité indépendant : propriété d'Embarcadero, l'éditeur est le seul à pouvoir décider de l'avenir et de l'ajout de nouvelles fonctionnalités au langage [1].

5.4 L'interface de développement

L'environnement de développement s'appuie sur un éditeur d'interface graphique associé à un éditeur de code source. Il doit son succès à sa facilité d'utilisation pour développer des applications graphiques et/ou liées aux bases de données. On l'a souvent comparé à *Visual Basic* de *Microsoft* pour

cette facilité de développement. On peut même dire que par un amusant mouvement de balancier et de personne, le VB influença Delphi qui à son tour influença par la suite le VB.

L'environnement de développement auto-génère du code pour faciliter le travail du programmeur. Il maintient une correspondance automatique entre la vue de conception (la fenêtre que le programmeur bâtit en déposant des composants graphiques) et l'éditeur de code (la vue affichant le code source qui créera ces composants à l'exécution).

Les données spécifiques aux composants sont stockées dans des fichiers d'extension **.DFM** alors que le code source en *Pascal Objet* est sauvegardé dans des fichiers d'extension **.PAS**. Alors que d'autres langages (comme *C#* avec Winforms) génèrent les instructions nécessaires à la création des composants de l'interface et l'injectent dans une section du code source du programme, Delphi sépare les données statiques de description des objets d'interface, à la manière de *XAML*, et se base sur des routines de la *VCL* pour relire et présenter l'interface lors de l'exécution.

L'interface de développement permet l'ajout de composants tiers (graphiques ou non) via un système de composants. La modularité est obtenue à la conception mais peut aussi être exploitée à l'exécution via un système de chargement dynamique de *paquets d'exécution*, *Borland* ayant étendu le concept de bibliothèques partagées et le format Windows *DLL* en introduisant un modèle propriétaire permettant d'enregistrer dynamiquement et d'exporter des classes entre modules. Le même système sera repris par *Microsoft* sous *Visual Basic* avec le format *VBX*, puis à l'échelle du système avec les composants *COM* et *ActiveX* [1].

5.5 Corpus de message

Tout d'abord, en utilisons le corpus SMS Spam Collection, ce corpus est un ensemble de messages étiquetés SMS qui ont été collectés pour la recherche sur le spam SMS. Il contient un ensemble de messages SMS en anglais de 5 574 messages, marqués comme étant **ham** (légitimes) ou **spam.** Elle est publique disponible à l'adresse suivante :

https://www.kaggle.com/uciml/sms-spam-collection-dataset

Nous avons choisi, d'une manière aléatoire, 40 messages (26 ham et 14 spam) pour la phase d'apprentissages et 20 autres messages pour la phase de test. Nous avons traduit ces messages de l'anglais en français.

Quelques exemples de messages sont présentés ci-dessous :

Messages spam :

INFO : Cela fait [29] jours que vos Cheques cash sont en attente de achter! Merci de nous contacter avant le 31 au 0899XXXXXX pourr retrait.

FELICITATIONS ! Vous avez été tiré au sort à 10h27 pour gagner cadeau SEJOUR pr 2 à l'ILE MAURICE ds l'Hotel XXXXXX 5* !! essai le 0899XXXXXX

Salut c moi ? J'attends toujours ton appel, a croire que je t'ai laisse mon numéro pour gagner100 auros pr rien, rappelle moi au 0899XXXXXX

Tirage au sort réalisé sous Controle d Huissier. Vous etes pour gagner du chéque No :345786 appel le 0899XXXXXX pour gagner.

Messages ham :

Je quitte ma maison maintenant.

Tu fais quoi Avez-vous eu à cette interview aujourd'hui?.

Nouvelle voiture et maison pour mes parents .

Il est complètement hors-forme : est également un déchet.

5.6 Eléments d'implémentation

Dans cette partie nous nous intéressons à la présentation de notre système en montrant quelques exemples de captures d'écran des différentes interfaces réalisées, afin de montrer les différentes fonctionnalités de notre application.

5.7 Interface principale

Au lancement de l'application, une fenêtre d'accueil s'affiche (Figure**4-1**). Cette fenêtre comporte un bouton de prétraitement de données d'apprentissage, un bouton d'exécution de l'une des méthodes de filtrage choisies et un bouton de fermeture de l'application.

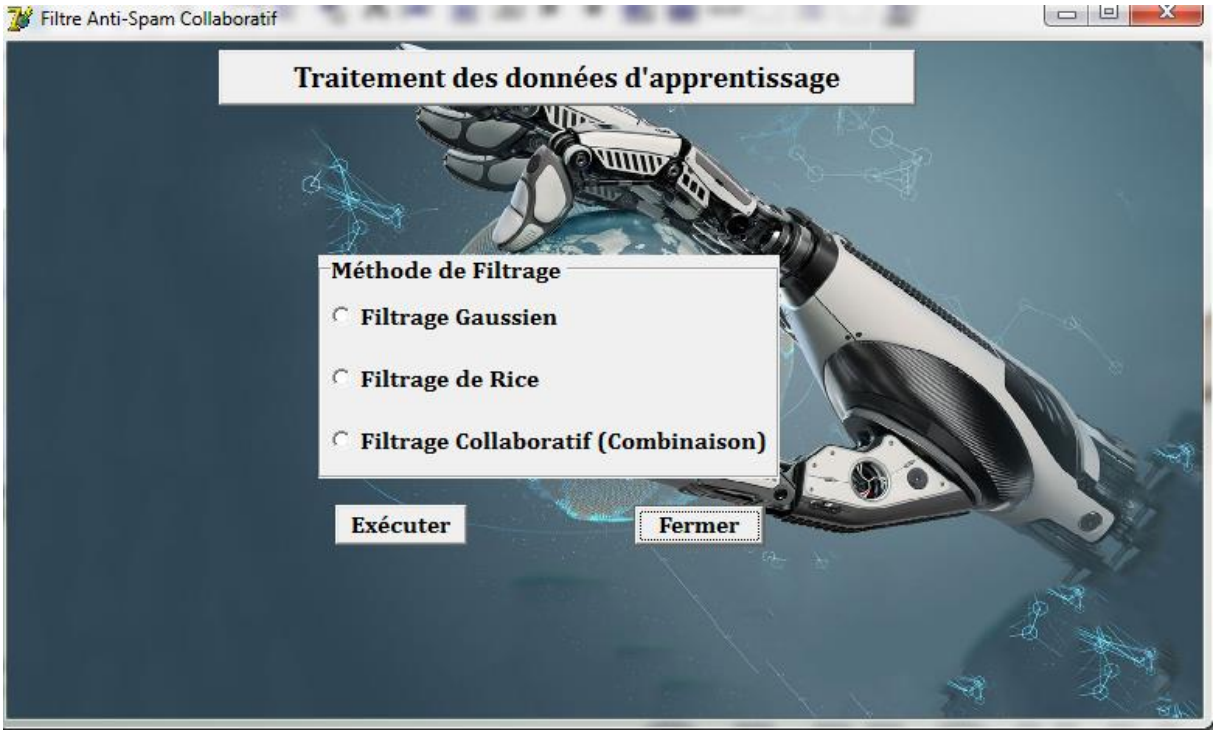


Figure 4-1 : Interface d'accueil

5.8 Traitement des données d'apprentissage

Une fois cliqué sur le bouton traitement des données d'apprentissage, une autre fenêtre s'affichera (Figure**4-2)**. Le prétraitement des messages est la première étape avant l'utilisation des techniques d'apprentissage automatique. L'ensemble des données doit être sous une forme compréhensible par le système d'apprentissage. Avant d'effectuer les expériences, nous avons prétraité les e-mails choisis en utilisons le corpus SMS Spam Collection (définit en haut). Le corpus de messagerie va passer par la phase de préparation ou prétraitement puis, nous avons utilisé les deux algorithmes d'apprentissage (Gaussien et Rice) pour construit un modèle qui permet de classer les messages à tester.

Pour effectuer le prétraitement des données, nous avons :

1-Compter le nombre de mots suspects parmi les mots suspects proposés.
2-Calculer le nombre total de messages.
3-Calculer le nombre total de mots pour chaque message.
4-Calculer le nombre de mots suspects se trouvant dans chaque message.

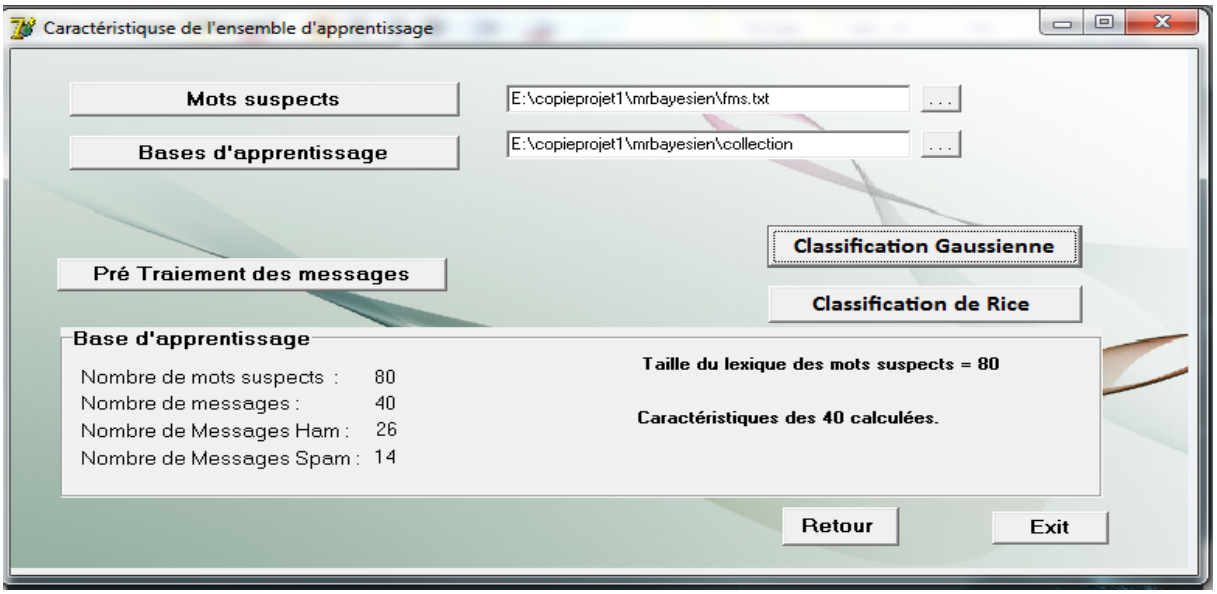


Figure 4-2 : Caractéristiques de l'ensemble d'apprentissage

5.9 La classification

La classification est la deuxième étape consiste à implémenter les deux classifieurs, Gauss et Rice. Pour cela nous avons deux boutons :
Cliquer sur le bouton classification Gausssienne, la fenêtre de la figure (4-3) s'affichera.
Ou bien cliquer sur le bouton classification Rice la fenêtre de la figure (4-4) s'affichera.

5.9.1 Classification Gaussienne

Cette fenêtre affiche les statistiques Gaussiennes : les moyennes, les écart-types, occurrences (le nombre total des spam et le nombre total des ham) et les fréquences (p(spam) et p(ham)) pour les deux variables *x* (le nombre total des mots dans chaque message) et *y* (le nombre total des mots suspects dans chaque message). Ainsi un bouton classification Gaussienne

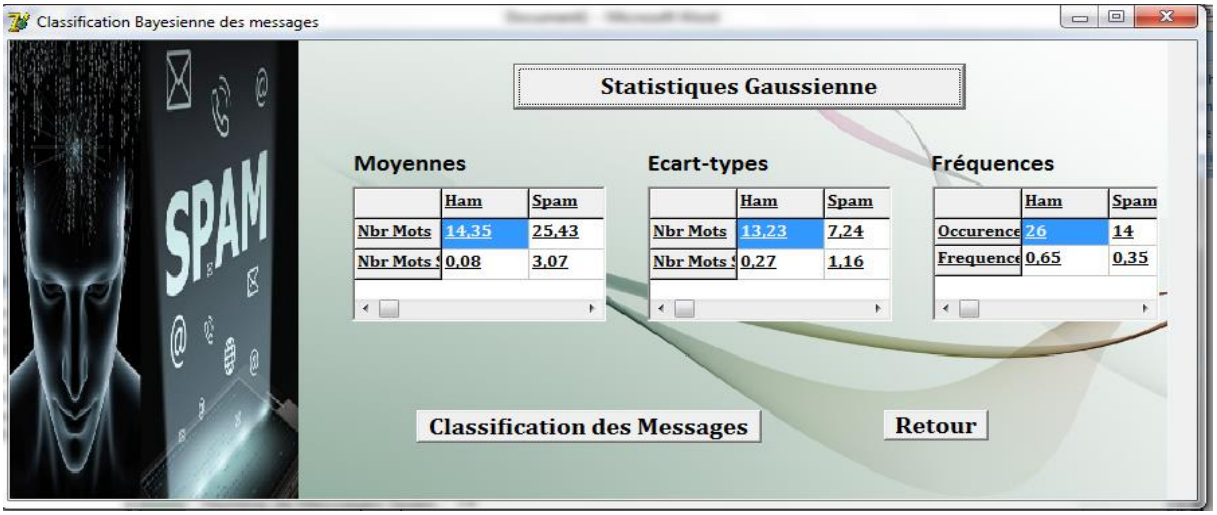


Figure 4-3 : Classification Gaussienne

5.9.2 Classification de Rice

Cette fenêtre affiche les statistiques Rice : les moyennes, les écart-types,les occurrences (le nombre total des spam et le nombre total des ham) et les fréquences (*p(spam)* et *p(ham)*) pour les deux variables *x* (le nombre total des mots dans chaque message) et *y* (le nombre total des mots suspects dans chaque message). Ainsi un bouton classification Rice.

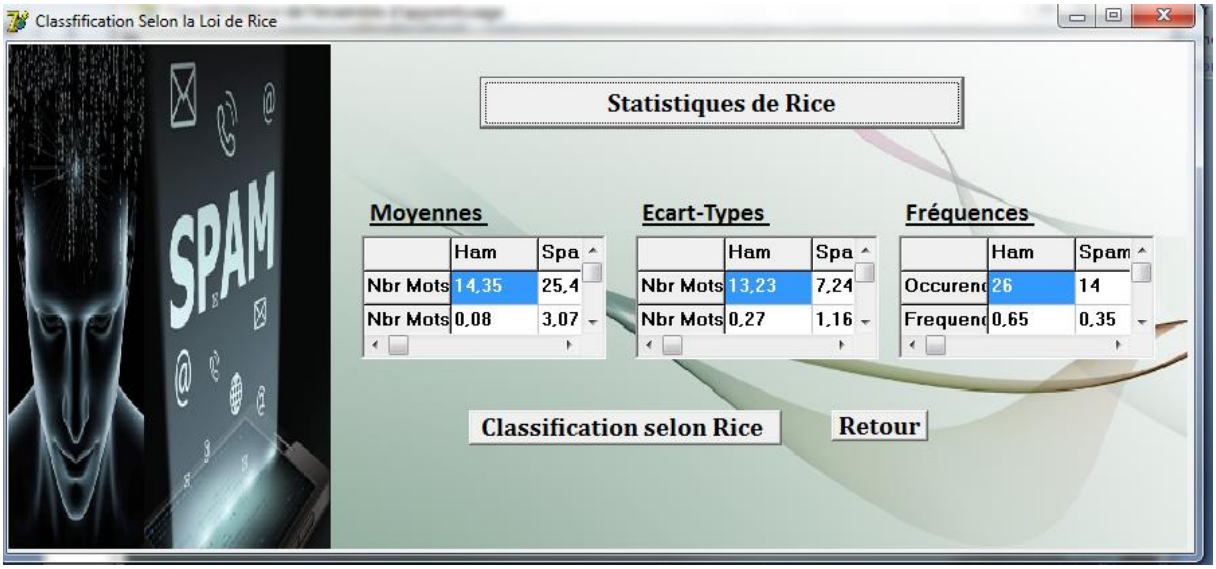


Figure 4-4 : Classification de Rice

5.9.3 Classification collaborative

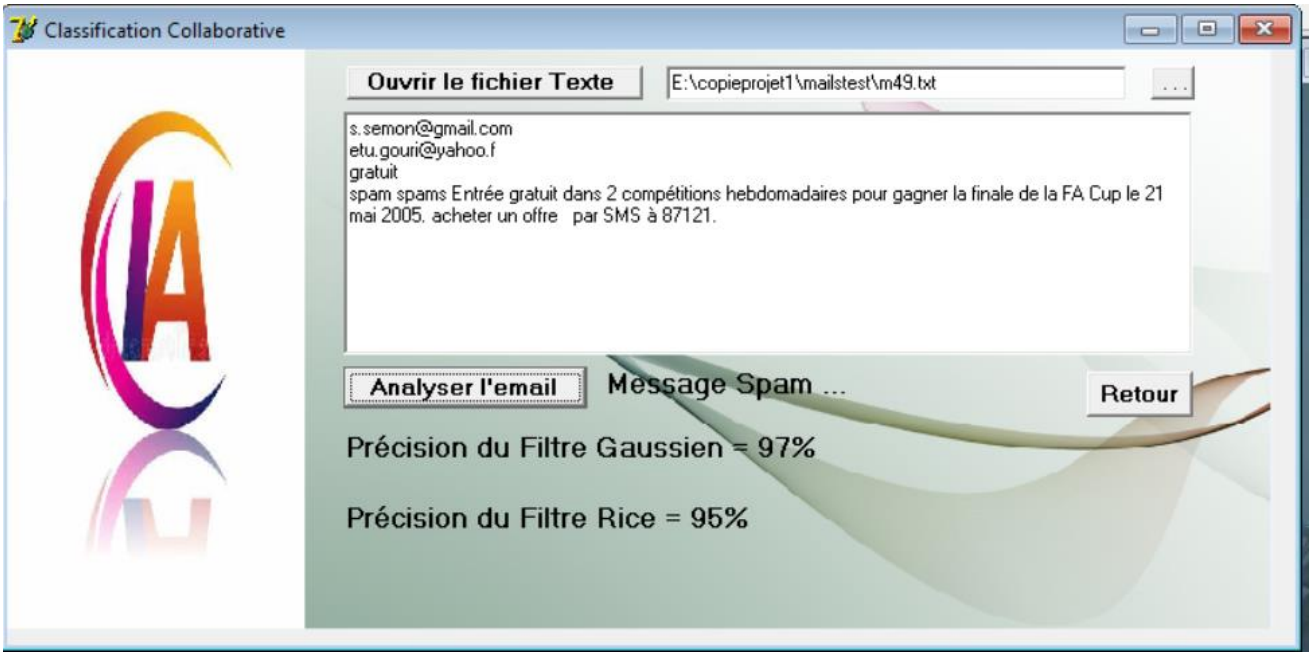


Figure 4-5 : Classification collaborative

Il s'agit de la fonctionnalité de détection de spam, qui consiste à classifier un courriel donné en Ham ou Spam en utilisant le filtre collaboratif. Pour des raisons de test, nous avons choisi d'introduire le message via l'interface de l'application en sélectionnant un courriel sauvegardé dans un fichier texte. Pour une application réelle, notre filtre doit être intégré comme module de détection pour un serveur de messagerie ou un client de messagerie.

5.9.4 Quelques exemples de test

Après la phase d'entrainement de notre modèle nous avons passé à la phase de test. Pour cela, nous avons choisi quelques exemples qui sont présentés ci-dessous :

Exemple d'un message spam

- Classification Gaussienne

Figure 4-6 : La fenêtre montre le test d'un e-mail classépar le classifieur Gaussien comme **un Spam**

- Classification Rice

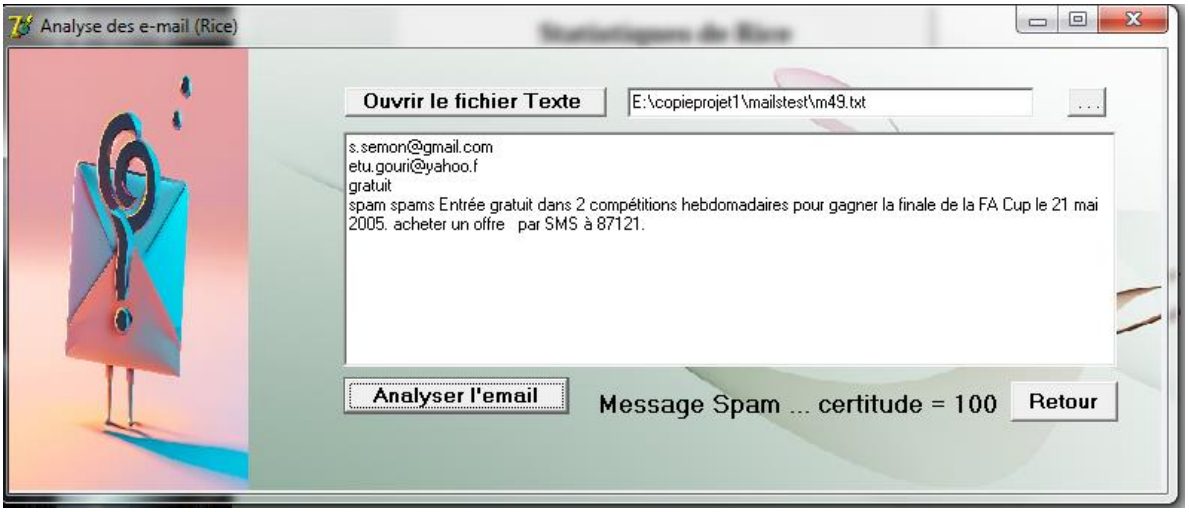


Figure 4-7 : La fenêtre montre le test d'un e-mail classépar le classifieur Rice comme **un Spam**

- Classification collaborative

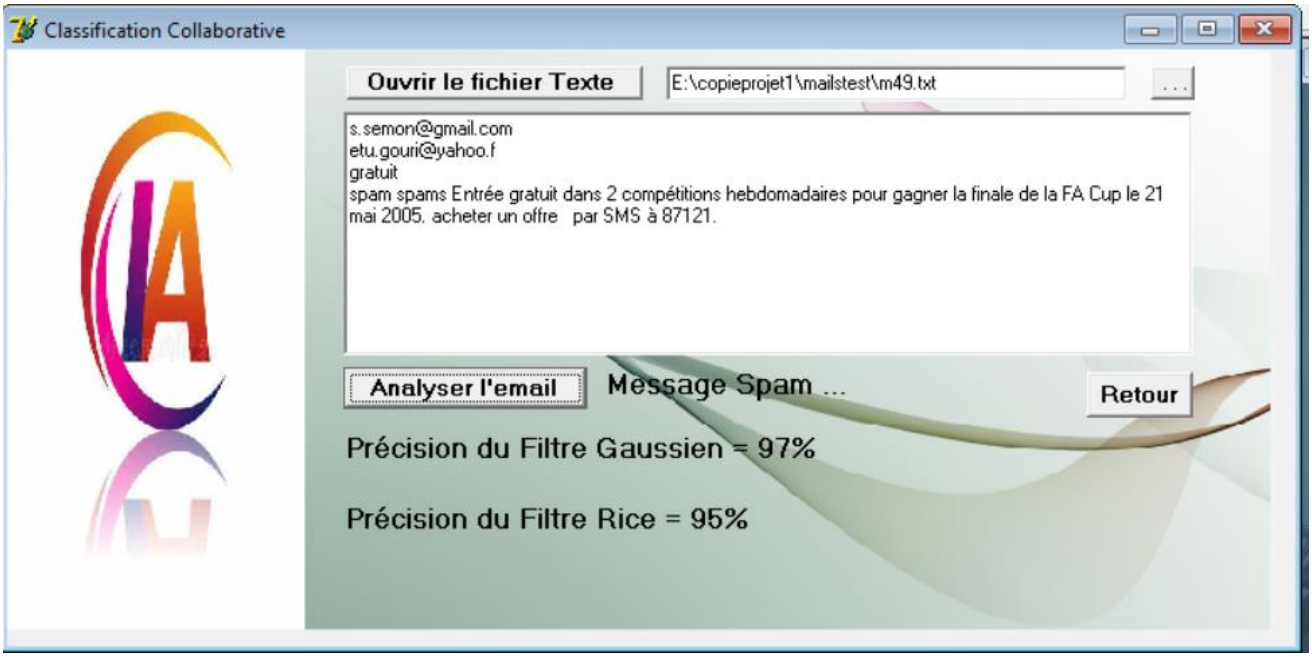


Figure 4-8 : La fenêtre montre le test d'un e-mail classépar le classifieur collaboratif comme **un Spam**

Exemple d'un message Ham

- Classification Gaussienne

Figure 4-9: La fenêtre montre le test d'un e-mail classépar le classifieur Gaussien comme **un Ham**

- Classification Rice

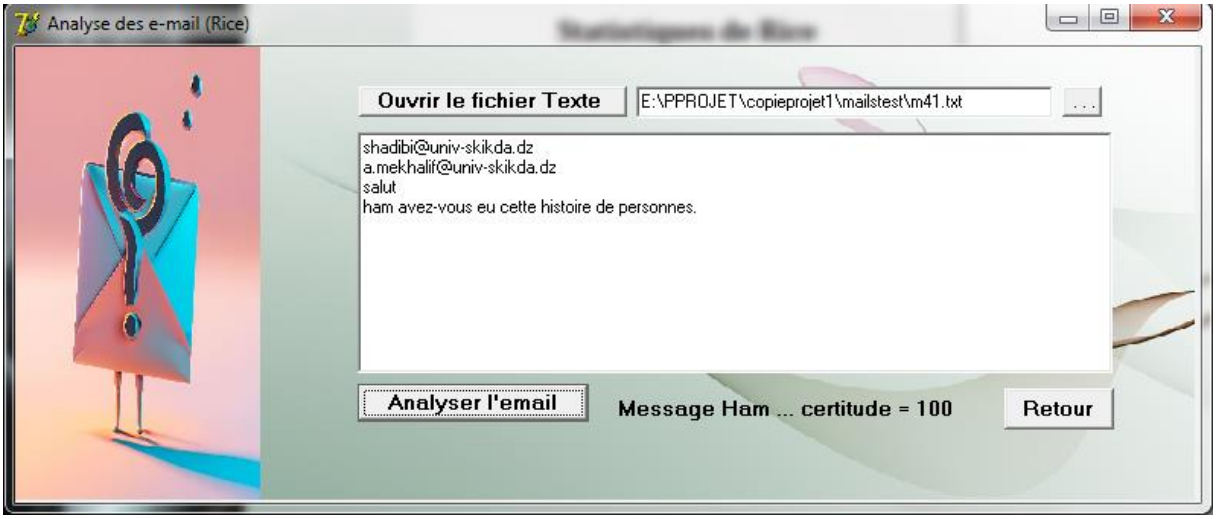


Figure 4-10 : La fenêtre montre le test d'un e-mail classépar le classifieur Rice comme **un Ham**

- Classification collaborative

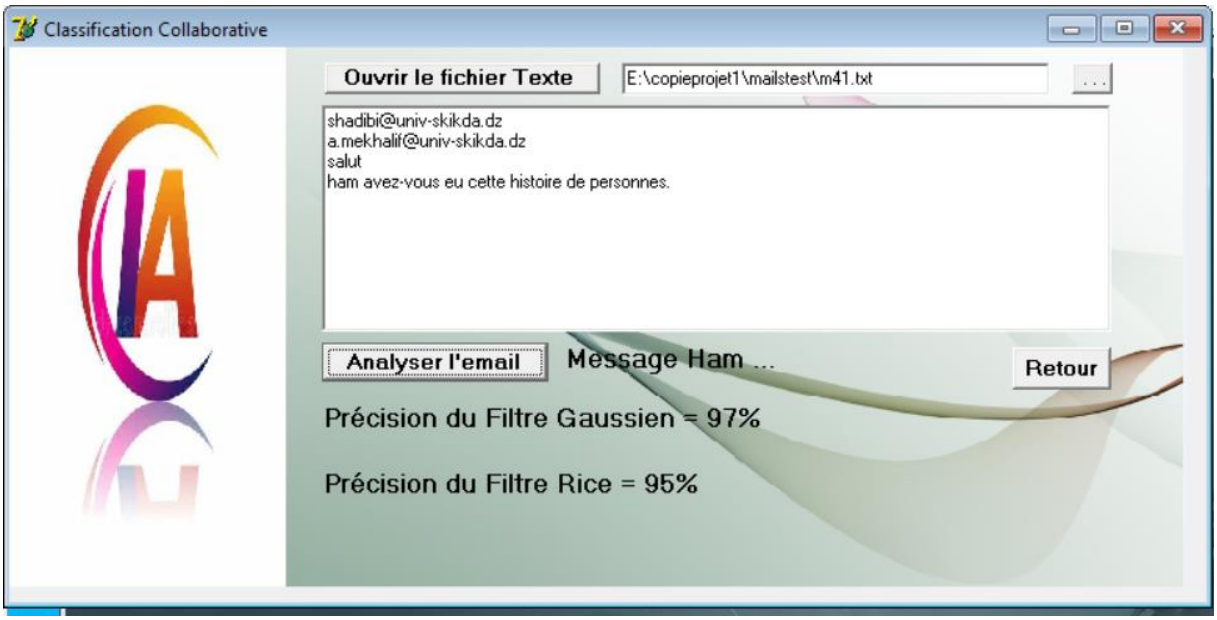


Figure 4-11: La fenêtre montre le test d'un e-mail classépar le classifieur collaboratif comme **un Ham**

Exemple de faux positif :

Un message "spam" ayant été étiqueté "ham" par le classifieur

- Classification Gaussienne

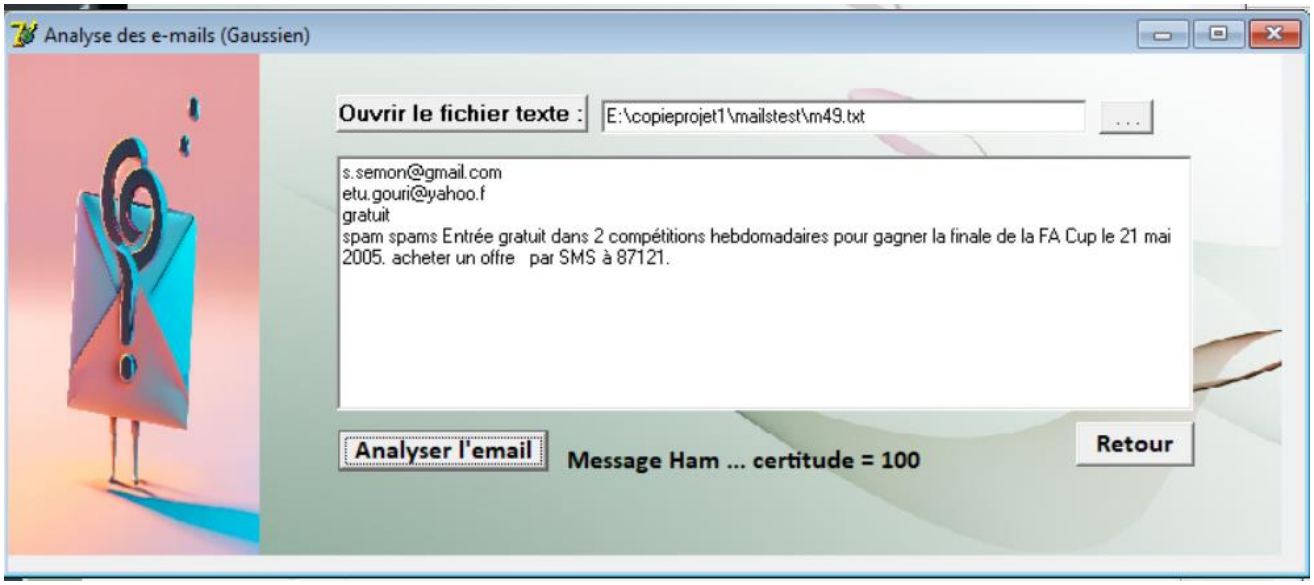


Figure 4-12 : La fenêtre montre le test d'un message "spam" mais classé "ham" par le classifieur Gaussien

- Classification Rice

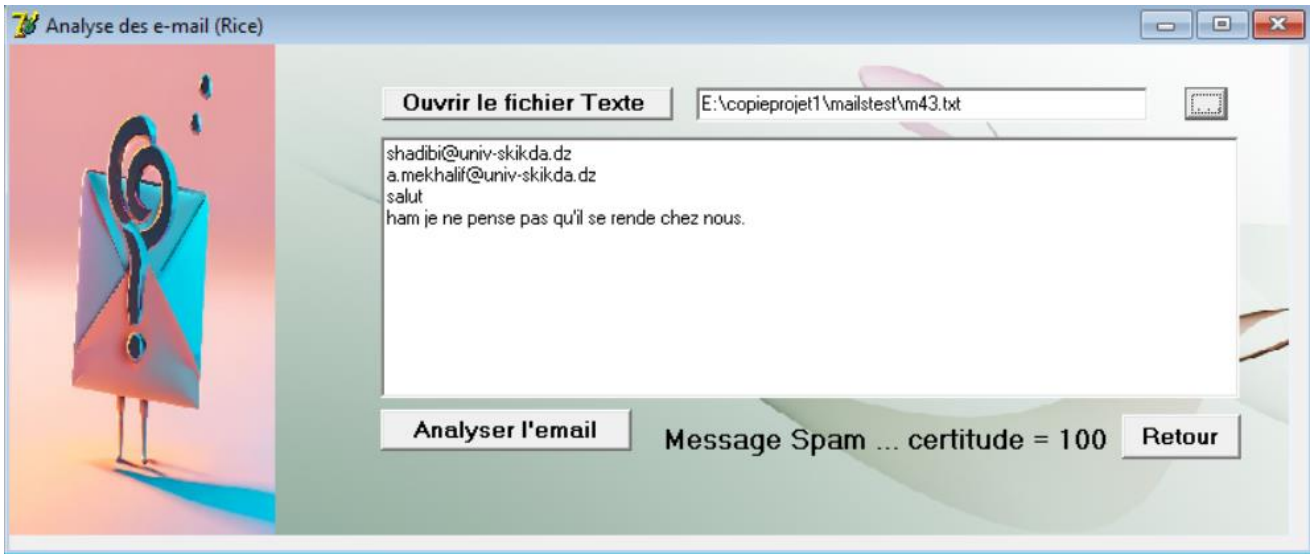


Figure 4-13 : La fenêtre montre le test d'un message "ham" mais classé "spam" par le classifieur Rice

- Classification collaborative

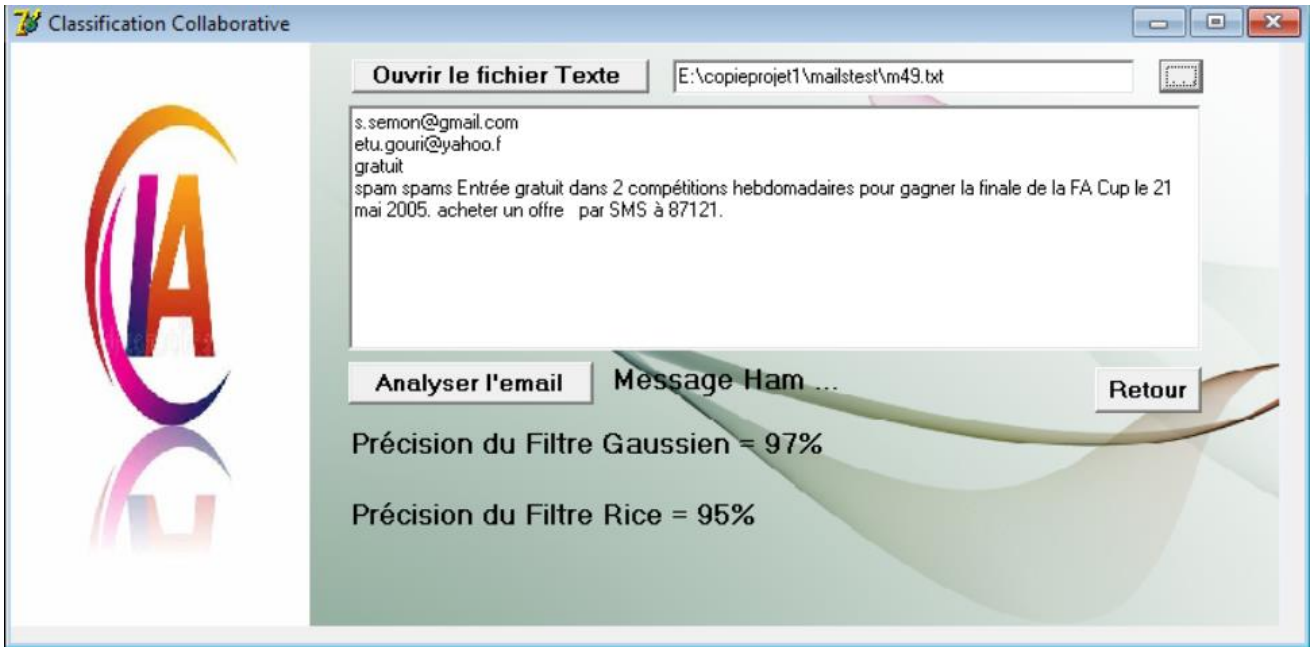


Figure 4-14 : La fenêtre montre le test d'un message "spam" mais classé "ham" par le classifieur collaboratif

Le tableau 2 présente les performances obtenues pour chaque classifieur pour 20 messages de test :

Filtre	Taux de bonne détection	Taux d'erreur
Gaussien	97%	3%
Rice	95%	5%
Collaboratif	95%	5%

Tableau 2 : les performances obtenues pour chaque classifieur

La coopération des deux méthodes nous a permis d'obtenir des bons résultats parce que la précision du filtre Gaussien est **97%** et la précision du filtre Rice est **95%**, les résultats sur le corpus utilisé sont satisfaisants. Les rare cas de faux positifs ou de faux négatifs sont dû à la précision des deux classifieurs individuels, et au corpus de message utilisé.
Il serait souhaitable d'élargir le corpus de message d'apprentissage, et aussi le lexique des mots suspect pour que le filtre collaboratif atteint des performances encore plus hautes.

5.10 Conclusion

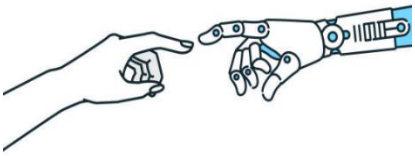
Ce dernier chapitre était consacré à l'implémentation et le test de notre filtre collaboratif pour la détection de courriels indésirables (spam). Nous l'avons commencé par l'introduction de l'environnement de développement, avant de passer aux interfaces de l'application, et qui donneront un bon aperçu sur son fonctionnement. La fin du chapitre était consacrée au test de notre filtre en commençant par la présentation du corpus des messages utilisé. Puis nous avons présenté quelques cas de catégorisation de courriels, avec détection de spam. Nous avons également montré le cas d'un faux positif est qui est dû à la faible quantité de données d'apprentissage.

6 Conclusion générale

L'avènement de l'intelligence artificielle dans la lutte contre le spamming marque une révolution dans la cybersécurité, offrant des perspectives inédites pour la détection, la prévention et la sensibilisation face à cette menace persistante.

À l'ère où les techniques de spamming deviennent de plus en plus sophistiquées, l'IA n'est plus un luxe mais une nécessité. Sa capacité à apprendre, à s'adapter et à réagir en temps réel aux nouvelles menaces la positionne comme un outil indispensable dans l'arsenal de la cybersécurité moderne.

L'IA transcende le rôle traditionnel des outils technologiques. Elle agit comme un partenaire stratégique, offrant non seulement une protection robuste contre le spamming, mais aussi une plateforme pour l'éducation et la sensibilisation des utilisateurs. En identifiant les comportements à risque et en proposant des formations personnalisées, l'IA joue un rôle crucial dans la construction d'une culture de sécurité informatique forte.



Avec chaque attaque détectée et chaque simulation effectuée, les systèmes d'IA deviennent plus intelligents et plus précis, renforçant ainsi continuellement nos défenses contre le spamming. Cette capacité d'apprentissage continu assure que les mesures de protection évoluent en tandem avec les menaces, garantissant une sécurité à long terme contre des attaques de plus en plus élaborées.

L'avenir de la cybersécurité réside dans une collaboration harmonieuse entre l'homme et la machine. Tandis que l'IA fournit la rapidité, l'efficacité et l'analyse de données, l'intervention humaine reste essentielle pour superviser, interpréter et prendre des décisions stratégiques basées sur les insights fournis par l'IA.

Cette synergie homme-machine est la clé pour une stratégie de cybersécurité vraiment résiliente.

En fin de compte, l'intégration de l'IA dans la lutte contre le spamming est un pas vers un avenir où la sécurité numérique est plus accessible, plus intelligente et plus réactive. Alors que nous continuons à naviguer dans un paysage numérique complexe et en constante évolution, l'IA se dresse comme un gardien vigilant, protégeant non seulement nos données mais aussi notre confiance dans la technologie qui façonne notre monde.

Dans ce travail de master, nous avons étudié deux modèles d'apprentissage automatiques basées les deux sur la règle de décision bayesienne, cependant le premier utilise la loi de Gauss et la seconde sur la loi de Rice. Nous avons également proposé une démarche de collaboration améliorer les résultats d'analyse des messages. En pespective à ce travail, nous commençons par tester notre systèmes sur d'autres bases d'apprentissage, et d'étendre notre étude pour des modèles d'apprentissage plus performnants, tels que les modèles profond.

Bibliographie

[1] :Wikipedia, https://fr.wikipedia.org 2019, 26/05/2024.09:44.
[2] : khattabi k Détection de SPAM par collaboration de classifieurs probabilistes 2019
[3] : Tutoriel sécurité Intrusions réseaux & attaques Web https://doc.lagout.org/network/Intrusion%20reseaux%20et%20attaques%20Web.pdf,Joel Winteregg - Massino Iritano 2006,28/05/2024, 11:37.
[4] : OUAZAR .O REDJDAL. L Conception et réalisation d'un système de messagerie interne Cas:"IFRI", L'université de bejaia 2015-2016.
[5] : DJOB.PN. Mise en place d'un système de messagerie électronique: Cas du fonds de prévoyance militaire,licence professionnelle en science informatique.ifpg - isfpt - ingénieur de conception réseaux et télécoms. 2008.
[6] : BOUCHE.S BUTEL.A CHAMON.E GONEL.JF BERGEROUN.R,MERTIN.M. Securité de la messagerie. clusif. 2005.P .
[7] : Guillon, "Etat de l'art du pourriel, solutions et recommandations " 10 décembre 2008.
[8]: G. Schryen, Anti-Spam Measures Analysis and Design, Berlin Heidel berg New York, Springer, 2010.
[9] : aidewindows, www.aidewindows.net/phishing.php, 28/05/2024, 14:40.
[10]: Gherabi.c Détection des spams se basant sur les techniques de classification, Thèse de Master, UNIVERSITE MOHAMED BOUDIAF - M'SILA2017 /2018.
[11]: B. Markines, C. Cattuto, and F. Menczer, "Social Spam Detection," presented at the 18th International World Wide Web Conference (W3C 2009), 2009.
[12]: Sophos. (2015, 2 mai). SPAMIONSHIP des douze pays émettant le plus de pourriels – quel rôle *VOUS* pouvez jouer Available: https://www.sophos.com/fr-fr/press-office/pressreleases/2014/10/dirty-dozen-spampionship.aspx 2019, 03/06/2024, 10:00.
[13] : J. M. G. H. Tiago A. Almeida , Akebo Yamakami, "Contributions to the study of SMS spam filtering: new collection and results," 2011.
[14] : Ekpao Anani, PASSIGUE , Analyse et détection de pourriels textuels dans les réseaux sociaux par apprentissage, UNIVERSITÉ DU QUÉBEC EN OUTAOU 18 Août 2015
[15] : T. O. a. T. White, "Immunity from spam : An analysis of an artificial immune system for junk email detection," in Proceedings, 4th International Conference, ICARIS, Banff, Alberta, Canada, 2005, pp. 276-289
[16] : A. C. A. Kołcz, "Lexicon randomization for near-duplicate detection with I-Match," Supercomput Springer Science+Business Media, LLC 2008, pp. 45: 255–276, 26 January 2008.
[17] : Théorème de bayes https://fr.wikipedia.org/wiki/Th%C3%A9or%C3%A8me_de_Bayes 2019, 20/06/2024, 15:00.
[18] : Fonction exponentielle, https://fr.wikipedia.org/wiki/Fonction_exponentielle,2019 16/06/2024, 15:30.
[19] : UML (informatique),https://fr.wikipedia.org/wiki/UML_(informatique), 2019; 21/06/2024, 10:00.
[20] : Delphi : définition et présentation de ce langage informatique (journaldunet.fr) ; 21/06/2024, 15:00.