RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université de Abderrahmane Mira - Bejaia
Faculté des Sciences Exactes
Département d'informatique

Mémoire de fin de Cycle
en vue de l'obtention d'un Master Professionnel
Option : ASR

Thème

Détection des spams basé sur Machine Learning

Cas: Spam Arabe

Réalisé par :
BAHOUCHE Ziane
BELHADAD Hicham

Membres du jury :

Examinateur	**: Dr. M. Sadi**	**M.C.B U.A/Mira Bejaia.**
Examinatrice	**: Dr. H. Khaled**	**M.C.B U.A/Mira Bejaia.**
Promotrice	**: Dr. L. Hamza**	**M.C.A U.A/Mira Bejaia.**

Promotion 2020/2021

- Remerciements-

En premier lieu nos plus sincères remerciements vont au bon Dieu tout puissant qui nous a donné la grande volonté et un savoir adéquat pour mener à bien ce modeste travail.

Nos remerciements sont adressés également à nos chers parents pour tous les sacrifices consentis à notre égard et leur énorme soutien.

Nos vifs remerciements, s'adressent à notre promotrice M^{lle}. HAMZA Lamia dont les conseils et orientations nous ont été précieusement utiles pour la réalisation de ce projet.

Nous remercions très respectueusement tous les membres de jury : M. Sadi Mustapha et M^{lle}. Dr Khaled Hayette qui ont accepté d'examiner notre travail.

Enfin, nos remerciements s'adressent à toutes les personnes ayant contribué de près ou de loin à la réalisation de ce travail.

- Dédicaces-

Je dédie ce modeste travail aux personnes chères à mon cœur.

À mes chers parents ma mère et mon père
ainsi qu'à mes frères et sœurs pour leur patience, leur amour,

leur soutien et leurs encouragements.

A tous mes amis et a tous mes camardes
Sans oublier tous mes professeurs
De l'enseignement supérieur
et a tout ceux qui m'ont aidé dans
l'élaboration De ce travail.

Merci …

Table des matières

Table des figures

Liste des tableaux

Liste des abréviations

C.T	**C**lassification de **T**exte
DTC	**D**ecision **T**ree **C**lassifier
FAI	**F**ournisseurs d'**A**ccès **à I**nternet
IDF	**I**nverse of **D**ocument **F**requency
K-ppv	**K p**lus **P**roche **V**oisin
KNN	**K- N**earest **N**eighbours
K-ppvn	**K-P**lus **P**roches **V**oisins
ML	**M**achine **L**earning
NLP	**N**atural **L**anguage **P**rocessing
NB	**N**aive **B**ayes
PTR	**P**ublic **T**est **R**ealm
RIR	**R**egional **I**nternet **R**egistry
RBL	**R**ealtime **B**lackhole **L**ist
RF	**R**andom**F**orst
SMTP	**S**imple **M**ail **T**ransfer **P**rotocol
SVM	**S**upport **V**ector **M**achine
TF	**T**erm **F**requency
TF-IDF	**T**erm **F**requency **I**nverse **D**ocument **F**requency

Introduction générale

Le courrier électronique (e-mail) est important dans notre vie quotidienne, qui est devenu largement utilisé par de nombreuses personnes, individus et organisations, et c'est un moyen rapide et économique pour échanger des informations. Dans le même temps, le courrier électronique est l'un des problèmes croissants et coûteux liés à Internet aujourd'hui, auquel cas il est appelé spam. Les courriers indésirables sont principalement marchands ou ont des liens attrayants vers des sites Web célèbres, mais ils mènent à des sites qui sont importuns.En conséquence, les e-mails de spam entraînent une diminution de la confidentialité, la propagation de virus, l'occupation de l'espace dans la boîte e-mail et la destruction des serveurs de messagerie. Par conséquent, l'utilisateur perd beaucoup de temps à filtrer les importations de courrier électronique et à annuler le courrier indésirable. La découverte des e-mails indésirables classe les e-mails comme spam ou non-spam (ham), donc ce processus est lié au problème de classification.

Les méthodes de classification peuvent être utilisées pour détecter les spams, mais l'ensemble de données contient généralement un grand nombre de caractéristiques triviales ou répétitives, ce qui réduira la précision de la classification. La sélection de fonctionnalités est la réponse à cette question, elle est utilisée pour sélectionner un sous-ensemble de toutes les fonctionnalités. Le but de notre projet est de réaliser un filtre anti-spam arabe basé sur ces méthodes de classification.

Développer des systèmes de classification de texte pour les documents Arabes en général et les e-mails arabes est une tâche difficile en raison de la nature complexe et riche de la langue arabe [21]. Il existe une grande communauté arabe d'utilisateurs utilisant des services de messagerie électronique. Ils ont besoin d'un filtre anti-spam arabe suffisant et puissant. Dans le domaine du spam arabe, de nombreux auteurs arabes ont travaillé sur le spam Web en arabe et ont publié de nombreux travaux [21], mais en fait le domaine du spam par courrier électronique en arabe est très pauvre, aucune recherche publiée. Le but est de trouver la solution optimale de la fonction objective, visant à minimiser la dimension de la ligne.

Pour cela, nous allons utiliser les méthodes les plus célèbres pour induire des classificateurs de spam tels que Machine à Vecteur de Support, Naïve Bayes et K Plus Proche Voisin, Arbre de Décision et évaluer les résultats du classificateur.

Dans notre étude, nous avons utilisé l'algorithme Arbre de Décision (*DecisionTreeClassifie)* pour la classification des spams, ce qui génère des résultats avec un faible taux d'erreur.

Guide du lecteur :

Ce mémoire est organisé de la manière suivante :

Dans le premier chapitre, nous présentons la définition de spam, à travers ses types, ses objectifs, ses contenus et ses impacts, ensuite les différentes techniques utilisées pour détecter les spams.

Dans le chapitre 2, nous présentons la définition de la classification, ensuite les étapes de classification de textes, et différents processus de classification permettant de présenter du texte et la pondération des termes, puis les techniques de classification qui présente les méthodes d'apprentissage automatique et les algorithmes d'apprentissage supervisé.

Dans le chapitre 3, nous présentons la définition de spam Arabe, ensuite nous présentons notre problématique et notre proposition, puis présentation de l'architecture de notre système, explication des étapes de construction et de validation d'un modèle de détection avant sa mise en production, finalement nous détaillons les méthodes d'évaluation des algorithmes de classification.

Dans le chapitre 4, nous présentons les outils et l'environnement de programmation utilisés, ainsi les bibliothèques essentielles pour l'apprentissage automatique, dans ce chapitre, nous allons mettre en clair les étapes de l'implémentation montrée dans l'architecture du modèle proposé.

Finalement, nous concluons par une conclusion et des perspectives et quelques travaux potentiels futurs.

CHAPITRE 1
Détection et filtrage des spams

1.1 Introduction

Ces dernières années, le besoin d'utilisation des adresses e-mail est devenu un mécanisme de communication de plus en plus indispensable. Par conséquent, la gestion des courriels est un problème important et croissant pour les individus et les organisations, car elle est sujette à des abus. L'affichage aveugle de messages électroniques non sollicités, appelés spam, est un exemple de mauvaise utilisation. Le spam est communément défini comme l'envoi de courriels de masse non sollicités, c'est un email qui n'a pas été demandé par plusieurs destinataires. Dans cette section, nous commençons par la présentation de l'historique et la définition des spams, ensuite nous présentons les différents types de spams, ainsi que l'explication des différentes manières de reconnaissances des spams, puis les impacts et les objectifs des spams, finalement nous détaillons les techniques de filtrages du spam qui conclut notre chapitre.

1.2 Spams

1.2.1 Historique de mot spam

Le mot SPAM provient de la contraction de l'expression "SPiced hAM" (jambon épicé), marque américaine déposée par la société Hormel Foods en 1937.

Le mot SPAM, comme on l'utilise aujourd'hui pour désigner un courrier indésirable, vient d'un sketch des Monty Python de 1970. Par la suite, il a été utilisé la 1ère fois en 1994 sur Internet par un internaute en colère pour dénoncer la première pratique de spam de l'histoire [1].

1.2.2 Définition :

Le spam est un courrier électronique envoyé massivement à de nombreux destinataires sans que ces derniers aient accepté de le recevoir. Souvent, la collecte des adresses électroniques figurant dans la base de données des spammeurs s'effectue illégalement. En d'autres termes, le spam est un courrier non sollicité que les destinataires ne s'attendent pas à recevoir [2].

1.3 Les différents types de spam

Le spam engendre un coût énorme aussi bien pour les particuliers que pour les entreprises. Il accapare non seulement leurs ressources informatiques mais également le temps des utilisateurs. Certains types de spam sont plus fréquents que d'autres.

1.3.1 Classification des spams

Les experts classent les spams selon les thèmes véhiculés. Dans la catégorie adulte, le spam propose des services ou des produits adaptés aux besoins des personnes majeures :

érotisme, conseils matrimoniaux, annonces du cœur, etc. le tableau 1.1 explique les catégories de classification des spams.

Remarque : Toutes les menaces politiques terroristes ou réalisées par des groupes extrémistes seront regroupées dans la catégorie spam politique.

Catégorie	Description
Financier	Le spam appartenant à cette catégorie incite les internautes à réaliser des prêts immobiliers, des investissements ou des achats à crédit.
Multimédia	Les spammeurs proposent aux internautes divers logiciels comme l'anti-virus, l'anti-spam, service d'hébergement et d'optimisation de site Internet, etc.
Escroquerie	Les internautes pourront voir les incitations à investir à l'étranger, les concours offrent aux usagers de gagner de l'argent facilement avec des jeux en ligne, la participation à un tirage au sort, des casinos en ligne, etc.
Education/Formation	Le spam propose des cours du soir, des formations de courte ou longue durée, des séminaires, etc. accessibles à un tarif intéressant.
objectif du spam	Le spam a pour mission de promouvoir certains produits et services non accessibles aux internautes dans le monde réel. Il ne tient pas compte de leur situation géographique puisque l'achat se fera en ligne. Par contre, les propositions des spammeurs varieront d'une saison à une autre pour obtenir le maximum de rendement : appareil de chauffage pour l'hiver par exemple et climatiseur en été.

Table 1.1 *Classification des spams.*

1.4 Reconnaissance des spams

Pour éviter les pièges et préserver vos données personnelles des cybercriminels, il est nécessaire de vérifier systématiquement ses principales composantes dès la réception d'un mail:

- le champ de l'expéditeur ;
- son objet, l'intitulé du message ;

- son contenu ;

1.4.1 Vérifiez l'adresse de l'expéditeur

Avant de cliquer sur un quelconque lien que l'e-mail pourrait contenir, vérifiez l'adresse e-mail de l'expéditeur. Ainsi, une adresse e-mail inconnue, avec des caractères spéciaux (ex : #&^*, etc.) ou d'un pays étranger doit vous alerter immédiatement !

1.4.2 Vérifiez l'objet du spam

Si le sujet du message contient des éléments suspects, curieux ou trop alléchant, il est fort probable que ce message soit un spam :

- un sujet commençant par "Re : …." pour faire croire à une réponse d'un tiers.
- un intitulé attrayant du type "Vous êtes l'heureux gagnant, etc.".
- une référence à un remboursement inattendu ou à un règlement de transaction qui vous est totalement inconnu.
- l'usage de caractères spéciaux, de symboles monétaires, de majuscules mal placées, etc.

1.4.3 Contenu du message

Après l'ouverture d'un e-mail suspect, il est probable que celui-ci soit un spam si vous repérez en son sein un, voire plusieurs des indicateurs suivants :

- de nombreuses fautes d'orthographes, l'utilisation de caractères spéciaux, majuscules et ponctuation inappropriées.
- des fausses promesses (gain d'argent ou de lot incroyable, retrouver l'amour perdu, etc.).
- des messages "humanitaires" vous demandant de faire suivre le message à vos contacts (chaîne de lettres).
- des messages vous demandant d'appeler un numéro de téléphone (en général surtaxé) ou de visiter un site qui s'avère payant.

1.4.4 Réaction après détection d'un spam

Si vous avez détecté un spam, il faut suivre les conseils suivants :

- ne cliquez sur aucun lien qu'il pourrait contenir.
- bloquez l'expéditeur du message, voire le domaine de l'expéditeur afin que les prochains messages de celui-ci soient déplacés dans le courrier indésirable. Il suffit généralement, de faire un clic avec le bouton droit de votre souris sur le message en cause, et de sélectionner une option du type "considérer comme spam" ou "courrier indésirable".

- assurez-vous d'avoir activé le service anti-spam fourni par ALGERIE TELECOM.

1.5 Impactes de spam

Le phénomène du spam ne cesse de croître ainsi que les victimes de ce véritable fléau. La pollution des messageries par le spam aurait atteint 65,26% en 2019. Autant pour les entreprises que pour les particuliers, les dangers du spam sont bel et bien présents et il est important de connaître l'impact de ces derniers [3].

1.5.1 Risques de sécurité

Une personne ne se souciant pas de la sécurité de son ordinateur peut être amenée à subir de graves conséquences [27] [28]. Les spammeurs ne peuvent pas accéder directement aux ordinateurs pour subtiliser les données personnelles. Toutefois, ils peuvent amener les utilisateurs à ouvrir des messages ou des pièces jointes contenant des logiciels malveillants, créant ainsi une brèche dans la sécurité du PC.

1.5.2 Atteinte à la vie privée

Le spam en lui-même n'est pas considéré comme un malware. Cependant, le spam peut contenir des malwares qui peuvent comprendre de multiples fonctions pour prendre possession de données confidentielles, et notamment celles permettant d'accéder aux sites web des banques. Par exemple, ils sont capables de faire des captures d'écran, de prendre le contrôle d'un ordinateur infecté puis d'envoyer du spam via ce PC, de connaître la localisation géographique de l'ordinateur.

1.5.3 Tromperie des consommateurs

Les arnaques sur Internet sont très fréquentes et utilisent beaucoup le spam comme outil de communication. L'utilisateur qui vient de recevoir du spam a de fortes chances d'être dirigé vers un site contenant des produits contrefaits ou qu'il ne soit victime d'hameçonnage (phishing). En pensant faire l'acquisition d'un produit ou service via ce site, il n'a fait qu'en réalité envoyer ses données confidentielles aux auteurs du spam.

1.5.4 Perte d'intérêt pour les technologies de l'information et de la communication

En raison de cette pollution virtuelle, le doute s'est véritablement installé chez les internautes initiés qui redoutent ce fléau. Voici un exemple : un internaute surfe sur un site avec des sujets qu'il trouve intéressants et souhaite être tenu au courant sur les prochains sujets qui seront ajoutés, ce dernier risque de ne pas donner son adresse email afin de recevoir les newsletters. La peur de recevoir du spam lui en a dissuadé.

Nous pouvons très bien imaginer que l'internaute pourrait utiliser une adresse email temporaire, mais cela est fastidieux et cette adresse, limitée dans le temps, ne pourra plus recevoir les newsletters futures.

1.5.5 Atteinte à la dignité humaine et à la protection des mineurs

Les adolescents n'ont jamais reçu une quelconque éducation sur les règles de comportements à appliquer vis-à-vis d'Internet et de ce fait, si l'ordinateur ne contient pas de filtrage de contenus [29] [30] [31], les plus jeunes encourent le risque de se connecter à des pages comportant des contenus choquants.

1.5.6 Surcoût engendré lors de la connexion

Suivant le coût de la connexion Internet, du nombre de courrier non sollicité reçu et du temps de téléchargement que cela prendra pour récupérer tous les messages, l'utilisateur peut rapidement arriver à un coût élevé. Sans compter la perte de temps a différencié les emails légitimes du spam.

1.5.7 Perte de productivité pour les entreprises

Les entreprises perdent en productivité dès lors que leurs employés passent du temps à télécharger et à supprimer le spam au lieu de se consacrer à leurs activités normales. Le coût peut varier en fonction du coût horaire de chaque employé. Aussi, les ressources des serveurs de messagerie, de la bande passante du réseau et de la capacité des périphériques chargés des copies de sécurité se consacrent au traitement et au stockage du courrier intempestif [3].

1.6 Objectifs des spams

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, les objectifs des spams sont très variés en voici une liste non exhaustive :

1.6.1 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 [4].

1.6.2 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 des idées politiques, culturelles ou religieuses et / ou d'organisations.

1.6.3 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 [5].

1.6.4 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* »,etc.

1.6.5 Malware

Est un logiciel conçu pour infiltrer ou endommager un système informatique [32] [33]. 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 malware est installé. Interdépendance entre spam et spam évolution des logiciels malveillants les logiciels malveillants de spam propagent les e-mails et les logiciels malveillants sont utilisés pour infecter l'hôte afin de contrôler à distance l'hôte et de l'utiliser pour envoyer plus de spam. 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 [5]. La figure 1.1 représente les statistiques de répartition des spams durant l'année 2019.

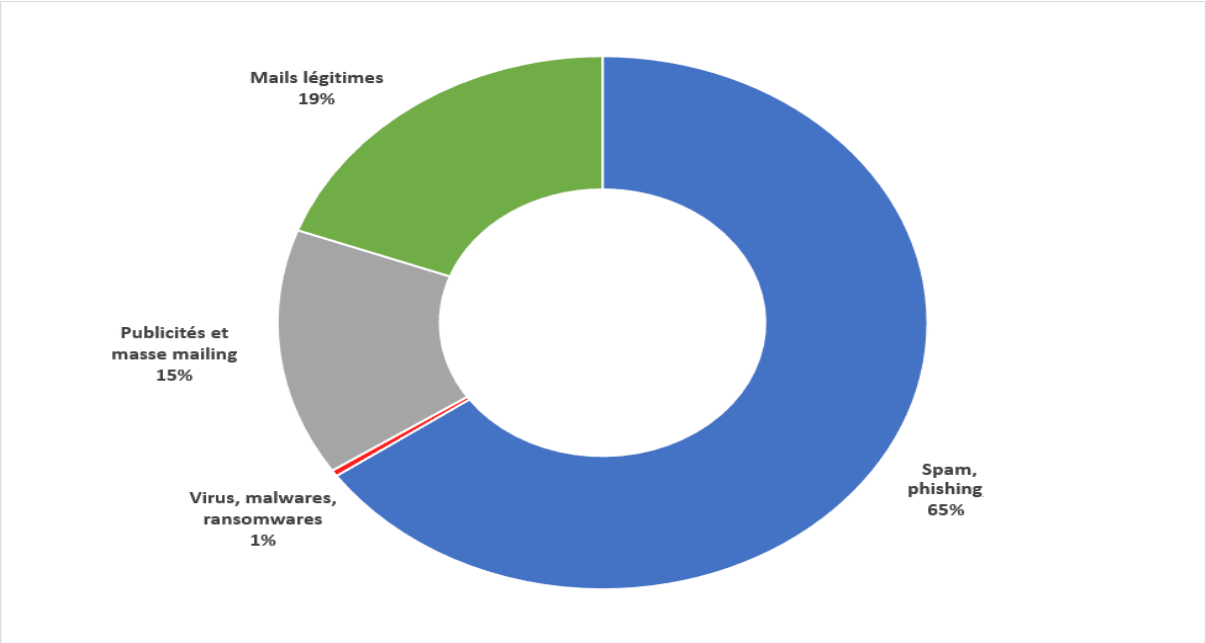

Figure 1.1 *Répartition des spams [35].*

1.7 Filtrage des spams

Même si les spams constituent un problème, il est aujourd'hui possible d'en limiter sensiblement les effets. Il suffit d'utiliser diverses technologies antispam de manière judicieuse. À l'heure actuelle, les solutions proposées s'appuient sur une combinaison de techniques permettant d'identifier les spams. Il n'existe pas de procédure standardisée pour filtrer un email, chaque utilisateur peut appliquer ses propres méthodes et surtout en combiner plusieurs différentes techniques. Dans cette partie on va parler sur l'emplacement des filtres, et les listes ainsi que les meilleurs techniques du filtrage des spams.

1.7.1 Emplacements des filtres

1.7.1.1 Filtres du côté de l'utilisateur

On peut citer deux types de filtres du côté de l'utilisateur :

- Le filtre antispam du client email : comme dans Outlook, Apple Mail ou Thunderbird, ce sont principalement des filtres bayésiens. Qui fonctionnent sur base des classements en spam effectués précédemment par l'utilisateur.
- Les règles créées par l'utilisateur : Si l'utilisateur a décidé de filtrer tous les emails provenant de votre nom de domaine afin de les placer dans sa poubelle, il sera presque impossible d'y échapper.

1.7.1.2 Filtres du côté serveur (webmails, FAI et entreprises)

- Les filtres maison : certaines organisations ont décidé de créer leurs propres systèmes de filtrage d'email. Dans ce cas, très peu d'informations seront disponibles afin de connaître les règles utilisées par ce type d'acteur.
- Utilisation de technologies disponibles sur le marché : qu'elles soient commerciales ou libres (comme le célèbre Spamassassin), ces technologies sont utilisées en combinaison avec d'autres afin de créer une solution anti-spam sur mesure.
- Les appliances : principalement utilisées en entreprise (mais pas uniquement), les appliances sont des serveurs "prêt à l'emploi" qui vont filtrer les emails entrants. Il suffit de les brancher en amont du serveur email de l'entreprise, et l'administrateur système n'a plus qu'à leur faire confiance.

1.7.1.3 Filtres du côté du routeur

Les filtrages du côté du routeur sont utilisés. afin de se prémunir de l'usage par les spammeurs de leurs plateformes, les solutions de routage d'email sont obligées de créer des filtres préventifs[6].

1.7.2 Listes

1.7.2.1 Liste noire

Une liste noire est un document rassemblant les emails des spammeurs et les adresses IP des serveurs qui ont déjà envoyé des Spams. Cette liste permet de refuser tous les emails qui sont envoyés à partir des adresses incluses dans cette liste noire.

1.7.2.2 Liste blanche

La liste blanche est l'opposé de la liste noire. Tout le monde est mis sur la liste noire par défaut à moins qu'ils soient spécifiquement stipulés sur la liste blanche et on ne reçoit seulement les emails qui sont envoyés par les personnes incluses dans cette liste blanche. Par exemple, on peut spécifier la liste de son carnet d'adresses comme liste blanche.

1.7.2.3 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 spams, les spammeurs ne se rendront pas compte qu'ils doivent émettre une validation afin que le message soit accepté.

1.7.3 Les techniques de filtrage des spams

1.7.3.1 RBL (Realtime Blackhole List)

RBL c'est une liste noire centralisée. Les listes RBL sont disponibles sur l'Internet via des serveurs qui collectent les adresses noires et les ajoutent dans leurs listes qu'ils partagent.

1.7.3.2 L'enregistrement DNS PTR

Logiquement, une adresse IP doit posséder un enregistrement de type (PTR) qui permet ainsi, via une requête DNS inverse (in-addr.arpa) de retrouver le nom d'hôte à partir de l'adresse IP. Tout serveur de messagerie doit posséder cet enregistrement. Il y a des années, on pouvait se baser sur cette information pour savoir si le relais SMTP qui nous envoyait un mail était « officiel » ou pas. Cependant, ce n'est pratiquement plus le cas aujourd'hui. Les fournisseurs d'accès à Internet et les opérateurs télécoms ne s'embêtent plus et créer systématiquement des enregistrements PTR pour l'ensemble de leurs adressages IP attribué par les (RIR).

1.7.3.3 L'authentification sur SMTP

L'authentification sur le protocole SMTP a été à l'origine pensé pour être une réponse au Spam, mais il s'est avéré n'être utile que pour identifier les expéditeurs. Par exemple, un fournisseur d'accès pourrait exiger une authentification de ses clients avant qu'ils envoient un mail. Cependant, les spammeurs sont autonomes et envoient leurs Spam eux-mêmes via leurs propres serveurs SMTP, des relais ouverts ou par des zombies.
On comprend vite que l'idée était bonne, mais inefficace devant l'autonomie des spammeurs.

1.7.3.4 Honeypot à Spam

Un piège à Spam est un pot de miel employé pour rassembler les Spam. Les pièges à Spam sont habituellement une liste d'adresse email fictive, ainsi, elles ne servent pas pour la communication, mais plutôt pour leurrer les spammeurs [7].

1.7.4 Filtres

1.7.4.1 Filtre basé sur les entêtes

Ce filtrage peut s'avérer très efficace. Le taux d'efficacité de ce type de filtre est d'environ 50%. Ce type de filtrage s'applique uniquement à l'entête du message et ne s'attarde pas au contenu du mail. Cette technique présente l'avantage de pouvoir bloquer les mails avant même que leur contenu ne soit envoyé. L'entête du message contient souvent assez d'informations pour pouvoir incriminer un mail. De plus, le taux de faux négatifs dans ce type de filtrage est quasiment nul. Etant donné que le contenu des messages n'est reçu que si l'entête est validé, alors cela apporte l'avantage de diminuer grandement le trafic sur le relais SMTP.

1.7.4.2 Filtre basé sur le contenu

Ce filtre est appliqué dans le cas où les entêtes n'ont pas révélé d'informations suffisantes pour considérer le mail comme un Spam. Le filtre basé sur le contenu permet donc d'analyser le corps des messages mail. Ce filtre est beaucoup plus sensible, car les informations sont subjectives.

Le contenu peut être composé de deux types d'informations qui sont du texte et des images. Cela laisse ainsi place à deux grandes familles d'analyses et d'algorithmes.

1.7.4.3 Filtre basé dur les 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 : sexe, sexy, 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 V.I.A.G.R.A. ou encore Vi a gra. Le danger de cette méthode est d'obtenir un nombre de faux positifs gênant. Prenons par exemple le cas du mot clé sexe, il peut très bien être utilisé légitiment dans le cas d'une demande d'information complémentaire à un candidat « Nom : Prénom : Sexe : ».

1.7.4.4 Filtrage statistique à base d'apprentissage

Par exemple le filtrage bayésien du spam est un système fondé sur l'apprentissage d'une grande quantité de spam et courriels légitimes afin de déterminer si un courriel est légitime ou non. Afin de bien fonctionner, le corpus de spam et le corpus de courriels légitimes doivent contenir idéalement plusieurs milliers de courriels [2].

1.7.4.5 Le filtrage par IP trusted

Une autre méthode mise en place pour catégoriser les services d'envoi en utilisant les données des utilisateurs qui vont par exemple mettre systématiquement telle ou telle newsletter dans la boite SPAM. Ainsi, vos adresses IP d'envoi auront une note et seront de confiance ou pas. Ce type de filtre est utilisé par Gmail, Hotmail et Yahoo mail par exemple.

1.7.4.6 Analyse heuristique

L'analyse heuristique constitue un ensemble de règles représenté sous forme d'expressions régulières. Elle permet d'étudier tous les mails, et trouver ceux dont les entêtes et le corps de message seraient susceptibles de correspondre à certaines caractéristiques.

1.7.4.7 Bases collaboratives de spams

Ces bases de signatures de spams sont utilisées de la même manière que les bases de signatures de virus. Elles sont alimentées par les utilisateurs de solutions anti-spam. La figure 1.2 illustre en quelques étapes le filtrage des spams.

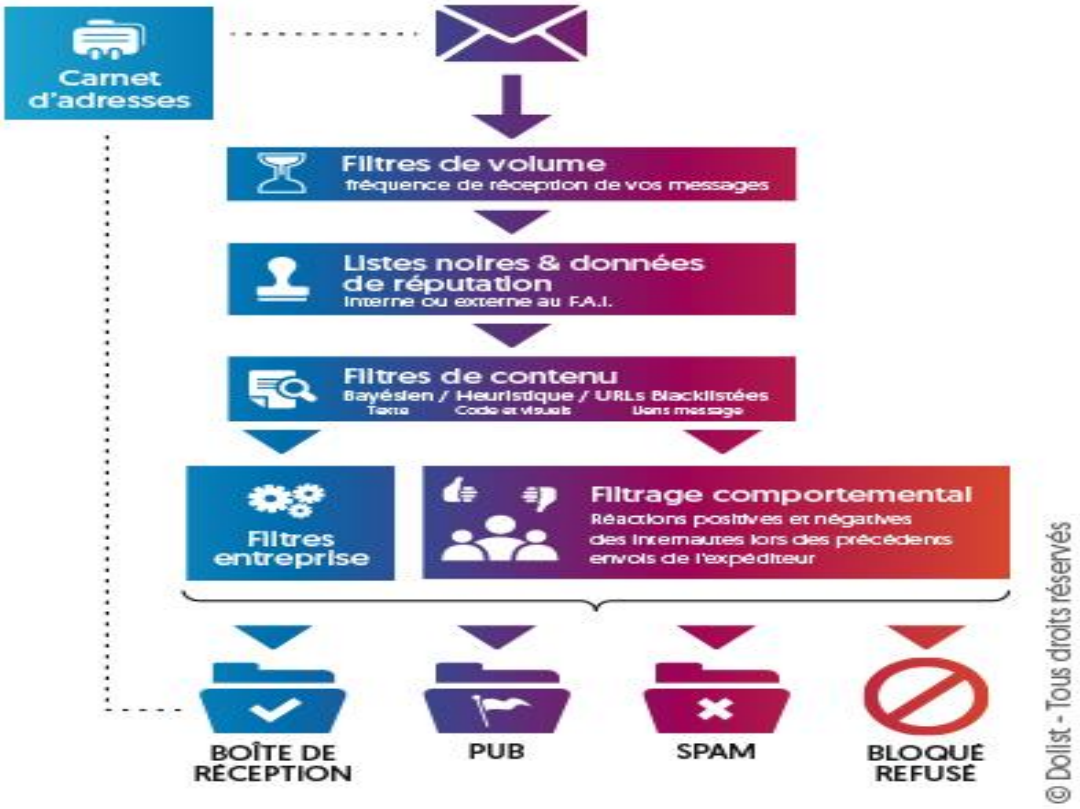

Figure 1.2 *Filtrage anti-spam [36].*

- Le 1er niveau tente d'évaluer la réputation de l'expéditeur (avant tout échange de message)
- Le 2éme niveau s'attache à analyser le contenu du message autant sur le plan technique.
- À la 3ème étape, prend la décision d'accepter ou non le message et la façon de le livrer

1.8 Conclusion

Dans ce chapitre, nous avons présenté la définition de spam, ainsi que les différents types de spam et les techniques qui nous permettent de reconnaitre un spam, les impacts qui présentent les risques de sécurité, les objectifs des spams, et les techniques utilisées pour filtrage des spams. Cela nous a permis de constater qu'il n'existe donc pas de solution parfaite, mais il faut appliquer un ensemble de méthodes et de principe afin de réduire au maximum les spams. Dans le chapitre qui suit nous présenterons la classification des textes et l'extraction d'information.

CHAPITRE 2
Classification des textes et extraction d'information

2.1 Introduction

La classification ou la catégorisions des textes est une tâche générique qui consiste à assigner une ou plusieurs catégories, parmi une liste prédéfinie, ou non à un document. C'est un problème qui a longtemps intéressé les chercheurs du domaine. Actuellement, la recherche dans ce domaine est toujours très pertinente, car les résultats obtenus aujourd'hui restent encore des sujets d`'améliorations. Dans cette section, nous commençons par présenter quelques définitions sur la classification, ensuite les étapes de classification de textes, ainsi que les différents processus de classification permettant de présenter du texte et la pondération des termes, puis les techniques de classification par lesquelles on va expliquer les méthodes et les algorithmes de classification, et enfin les problèmes spécifiques aux textes lors de l'apprentissage automatique, on conclut notre chapitre.

2.2 Définition

Plusieurs définitions du C.T (classification de texte) ont été vues depuis leur apparition.

Nous citons les deux définitions suivantes :

- **Définition1** : La C.T est une relation bijective qui consiste à "chercher une liaison fonctionnelle entre un ensemble de textes et un ensemble de catégories (étiquettes, classes)".
- **Définition2** : La C.T est le processus qui consiste à associer une valeur booléenne à chaque paire (dj, ci) _D ×C, où D et C présentent l'ensemble des textes et des catégories, respectivement. La valeur V(Vrai) est alors associée au couple (dj, ci) si le texte dj appartient à la classe ci tandis que la valeur F (Faux) est associée dans le cas contraire. Le but de la classification des textes est de construire une procédure (modèle, classificateur) notée : $\Phi : D \times C \rightarrow \{V, F\}$ qui associe une ou plusieurs étiquettes à un document dj [8].

2.3 Étapes à suivre pour classifier un texte

Pour effectuer l'opération de la classification des textes que nous avons définie, les étapes de la méthode standard de la classification automatique des textes peuvent être résumées comme suit :

- **L'apprentissage :** il comporte plusieurs étapes et en dérive un modèle prédictif :
  - **a)** nous avons un ensemble de textes étiquetés (pour chaque texte nous connaissons sa classification).

  b) à partir de ce corpus, nous extrayons les k descripteurs (mots, termes) (t1; t2; ..; tk) les plus pertinents en termes de problème à résoudre.

- **Le classement** : le classement d'un nouveau texte dx, qui comprend deux étapes :

  a) la recherche puis la pondération des occurrences (t1; ...; tk) des termes dans le texte dx a classé.

  b) application d'un algorithme d'apprentissage sur ces occurrences et prédire l'étiquette de ce texte dx [9].

2.4 Processus de classification de textes

Le processus reçoit le document texte d'entrée pour trouver sa classification, pour laquelle plusieurs étapes doivent être suivies. Ces étapes sont :

1) Représentation des textes ;

2) Pondération des termes ;

3) Réduction de la taille du vocabulaire ;

4) Choix de classificateur ;

5) Évaluation du modèle.

La figure 2.1 résume le processus de catégorisation des textes qui comportent deux phases : l'apprentissage et le classement.

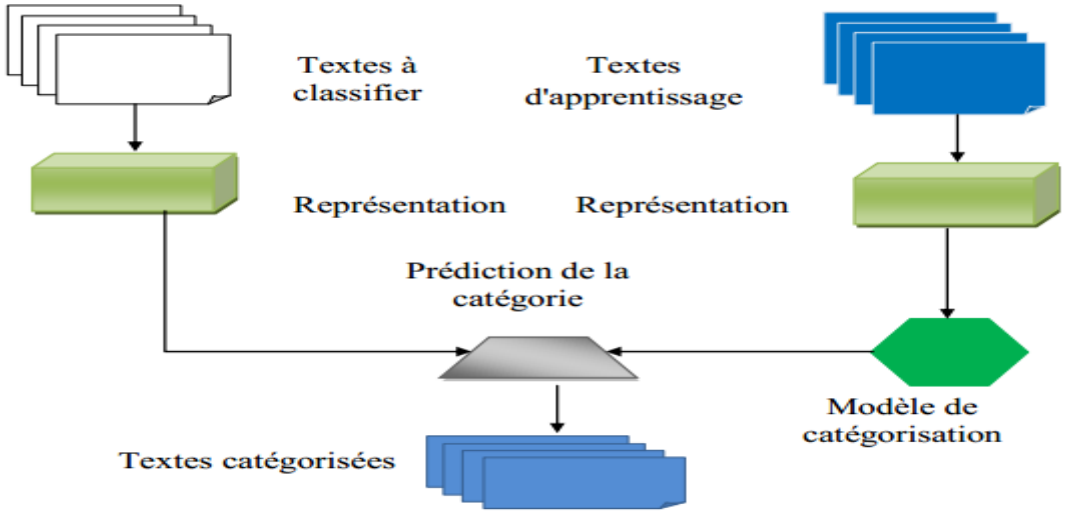

Figure 2.1 *Processus de classification de textes [10].*

2.4.1 Représentation des textes

La représentions des textes est une étape très importante dans le processus de C.T, pour cela il est nécessaire d'utiliser une technique de représentation efficace qui permet au texte d'être représenté sous une forme lisible par la machine. La représentation la plus couramment

utilisée est celle du modèle vectoriel [11] dans laquelle chaque texte est représenté par un vecteur de *n* termes pondérés.

Le document est alors transformé de sa version textuelle en une matrice [Document × Terme] comme présenté dans la figure 2.2.

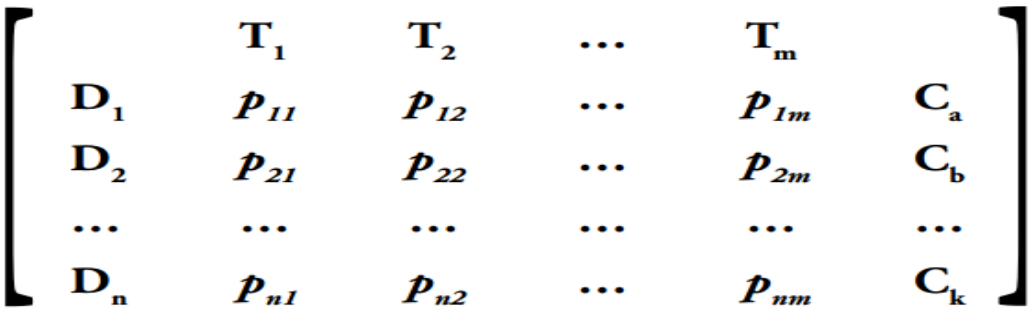

Figure 2.2 *Matrice Document × Terme [11].*

Chaque entrée représente un vecteur de termes où ***pnm*** est le poids du terme $\mathbf{T_m}$ dans le document $\mathbf{D_n}$ et $\mathbf{C_i}$ est la classe attribuée au document $\mathbf{D_i}$.

Les différentes méthodes qui existent pour la représentation des textes sont :

2.4.1.1 Représentation en sac de mots (bag of word)

Cette méthode consiste à représenter le document sous forme d'un vecteur de mots. Le processus de conversion du texte d'un document en un ensemble de termes est appelé analyse lexicale qui permet d'identifier et reconnaître les espaces de séparation des mots, les ponctuations, les chiffres, etc., pour qu'ils seront tous supprimés de la représentation. Cette représentation a comme avantage d'exclure toute analyse grammaticale et toute notion de distance entre les mots, mais présente les inconvénients suivants :

- les mots composés allemands peuvent être très complexes, exemple : *Levensversicherungsgesellschaftsangestellter (employé d'une société d'assurrance vie).*
- le Chinois et le Japonais ne séparent pas les mots par des espaces, ce qui peut mener à plusieurs segmentations.
- l'Arabe et l'Hébreu sont écrits de droite à gauche, mais certains éléments tels que les nombres sont écrits de gauche à droite [10].

2.4.1.2 Représentation avec les racines lexicales

Cette méthode consiste à remplacer les mots du document par des racines et à recombiner les mots ayant les mêmes racines en un seul composant. Par conséquent, plusieurs mots du document seront remplacés par la même racine. Cette méthode peut être réalisée en

utilisant l'un des algorithmes les plus connus de la langue anglaise, à savoir l'algorithme normalisé de Porter des mots, qui sert à les supprimer, et à les affixer pour obtenir la forme canonique. Cependant, la conversion automatique des mots à la racine entraînera certaines anomalies. En fait, pour les mots avec des significations différentes (par exemple, les mots jour, journal, journée ont la même racine «jour» mais ils sont des concepts différents), la racine peut être très courante, et cette représentation dépend également de la langue utilisée [17].

2.4.1.3 Représentation avec les lemmes

La lemmatisation consiste à utiliser l'analyse grammaticale pour remplacer les verbes par des formes infinitives et les noms par des formes singulières. En fait, un mot donné peut avoir différentes formes dans le texte, mais leur signification reste la même. Par exemple, les mots enseignés, enseignement, enseignée, enseigner, etc. Ces mots sont considérés comme des termes différents, alors qu'il s'agit de même racine enseigne. Ce type de représentation est simple, mais il peut entraîner la perte d'informations contextuelles nécessaires pour distinguer les termes polysémiques (à multiples significations) et l'existence de synonymes, même s'ils renvoient au même concept, ils sont considérés comme des termes différents.

2.4.1.4 Représentation avec les n-grammes

Cette méthode consiste à utiliser des n-grammes pour représenter des documents. n-gramme est une séquence de n caractères consécutifs, il s'agit de diviser le texte en séquences de n caractères en déplaçant une fenêtre d'un caractère.

Cette technique présente plusieurs avantages. n-grammes capture automatiquement les racines des mots les plus couramment utilisés sans passer par l'étape de recherche des racines, et c'est une méthode indépendante de la langue [13].

2.4.1.5 Représentation par phrases

Un certain nombre de chercheurs proposent d'utiliser les phrases comme unité de représentation au lieu des mots comme le cas dans la représentation « sac de mot », puisque les phrases sont plus informatives que les mots seuls, par exemple « recherche d'informations », « world wide web », ont un degré plus petit d'ambiguïté que les mots constitutifs, et aussi que les phrases ont l'avantage de conserver l'information relative à la position du mot dans la phrase. [14].

2.4.1.6 Représentation conceptuelle

La représentation conceptuelle sert à représenter le document sous forme d'un ensemble de concepts, ces derniers peuvent être capturés en utilisant les réseaux sémantiques prenons

l'exemple de la figure 2.3 en peut regrouper les trois termes (cime, sommet, crête) du vecteur dans le concept pic.

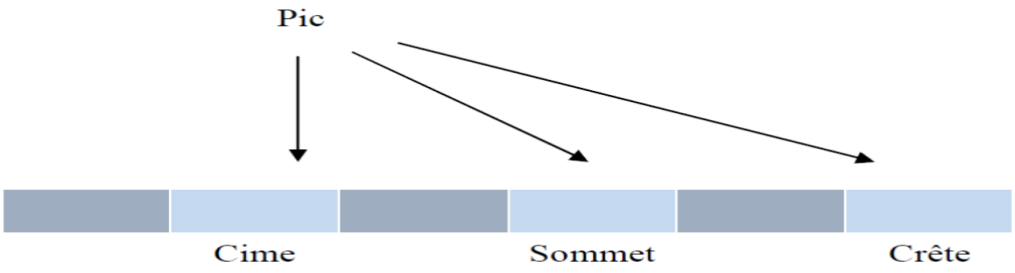

Figure 2.3 *Représentation conceptuelle du mot « pic ».*

L'avantage de cette méthode est qu'elle réduit l'espace de présentation car les mots synonymes ont le même concept. Cependant, l'inconvénient majeur de cette représentation est qu'il n'existe pas des bases lexicales pour toutes les langues.

2.5 Pondération des termes

La pondération des termes mesure l'importance d'un terme dans un document. Cette importance est généralement calculée sur la base de considérations et d'interprétations statistiques. Le but est de trouver le terme qui représente le mieux le contenu du document. Afin de calculer la pondération, nous distinguons les méthodes suivantes :

2.5.1 Mesure TF (Term Frequency)

Cette mesure est proportionnelle à la fréquence du terme dans le document (pondération locale). Elle peut être utilisée telle quelle ou selon plusieurs déclinaisons (log(tf), présence/absence, etc.).

2.5.2 Mesure TFIDF (Term Frequency Inverse Document Frequency)

Elle a été introduite dans le cadre du modèle vectoriel, elle donne beaucoup d'importance aux mots qui apparaissent souvent à l'intérieur du même texte, ce qui correspond bien à l'idée intuitive que ces mots sont plus représentatifs [2].

- **idf (Inverse of Document Frequency) :**

idf = log(N⁄Df)

Où :

Df: Le nombre de documents contenant le terme.

N : Le nombre total de documents de la base documentaire.

Les termes qui apparaissent fréquemment dans la base de documents ne devraient pas avoir le même impact que les termes moins fréquents. La mesure TFIDF est une bonne approximation

de l'importance du terme dans le document, particulièrement dans les corpus de documents de taille homogène. La mesure TFIDF est calculé comme suit :

$(T, D) = (T, D)$. log (N⁄Df(T))

Où :

TF(T, D) : la fréquence du terme dans le document.

2.5.3 MesureTFC

Le codage TF × IDF ne corrige pas la longueur des documents. Pour cette raison, le codage TFC est similaire au codage TF × IDF, mais il utilise la normalisation cosinus, pour ne pas favoriser les documents les plus longs [2].

$$TFC = (tk, d) = \frac{TF \times IDF(tk, d)}{\sqrt{\sum|r|\ TF \times (ts, d))2}}.$$

2.6 Techniques de classification

2.6.1 Méthode d'apprentissage automatique

Les systèmes d'apprentissage automatique peuvent être classés en fonction de l'importance et de la nature de la supervision qu'ils requièrent durant la phase d'entraînement. Nous s'intéressants aux quatre catégories principales : l'apprentissage supervisé, l'apprentissage non supervisé, l′apprentissage semi-supervisé, et l'apprentissage avec renforcement. Le tableau 2.1 exposer quelques différences entre les types d'apprentissage :

	APPRENTISSAGE SUPPERVISE	**APPRENTISSAGE NON-SUPERVISE**	**APPRENTISSAGE PAR RENFORCEMENT**
DEFINITION	L'algorithme apprend à partir de données labellisées	L'algorithme est entrainé à partir de données non labellisées sans indications particulières	L'algorithme interagit avec son environnement en réalisant des actions et en apprenant de ses erreurs et succès
TYPE DE PROBLEME	Régression et classification	Association et clustering	Basés sur un système de récompense
TYPR DE DONNEES	Données labellisées	Données non labellisées	Pas de données fournies au préalable

APPROCHE	Etudie les relations sous-jacentes qui lient les données en entrée au label	Découvre les motifs communs au sein des données d'entrée	Apprend une stratégie de comportement en fonction d'expériences passées et des récompenses perçues

Table 2.1 *Comparaison entre les différents types d'apprentissage.*

2.6.1.1 Apprentissage supervisé

Il s'agit de la forme d'apprentissage automatique la plus largement utilisée dans la pratique. Un algorithme supervisé est un algorithme à qui on présente l'entrée et la sortie (ou la cible) désirée en supposant qu'il y a une relation inconnue mais réelle entre les deux. Il doit minimiser l'erreur entre la sortie souhaitée et la sortie qu'elle génère. Ils sont généralement utilisés pour des problèmes de reconnaissance. Autrement dit, dans l'apprentissage supervisé, les données d'entraînement que vous fournissez à l'algorithme comportent les solutions désirées, appelées étiquettes (en anglais, labels) [15].

2.6.1.2 Apprentissage non supervisé

Les algorithmes dits non supervisés ne sont pas entraînés par le data scientist. ils s'appuient sur des méthodes d'apprentissage en profondeur pour reconnaître des modèles en combinant des ensembles de données d'entraînement non étiquetés, puis en observant les corrélations. Les modèles entraînés avec cette méthode ne sont pas dirigés pour trouver un résultat ou identifier des données en particulier.

2.6.1.3 Apprentissage semi-supervisé

Dans l'apprentissage semi-supervisé, l'ensemble de données contient des exemples étiquetés et non étiquetés. Le nombre d'exemples non étiquetés est beaucoup plus élevé que le nombre d'exemples étiquetés. Le but de l'algorithme d'apprentissage semi-supervisé est le même que celui de l'algorithme d'apprentissage supervisé. L'espoir ici est qu'en utilisant de nombreux exemples non étiquetés, un algorithme d'apprentissage peut trouver (on pourrait dire «produire» ou «calculer») un meilleur modèle [16].

2.6.1.4 Apprentissage par renforcement

L'algorithme d'apprentissage par renforcement est basé sur un système de récompense et de punition. L'algorithme se voit attribuer un objectif et essaie de se rapprocher de cet objectif afin d'obtenir le rendement maximal. Il se base sur des informations limitées et apprend de ses actions précédentes. Ces algorithmes peuvent dépendre d'un schéma (un modèle) ; ils doivent

alors suivre des étapes prédéfinies et le nombre d'erreurs et d'essais est limité. D'autres ne se reposent pas sur un schéma et expliquent à chaque fois qu'ils essaient [22].

2.6.2 Algorithmes d'apprentissage supervisé

Dans le cas de la classification de textes, l'apprentissage supervisé consiste à apprendre à partir d'exemples une fonction de prédiction. Cette fonction permettra par la suite de prédire la classe(ou la catégorie) de chaque nouvel objet (ici le texte). La figure 2.4 présente le principe de l'apprentissage supervisé dans le cas de filtrage de spam.

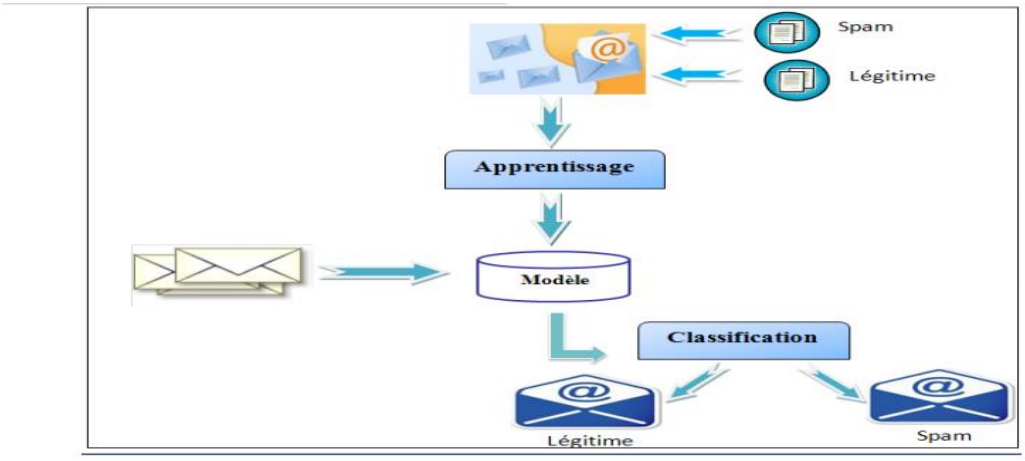

Figure 2.4 *Filtrage de spam à base apprentissage supervisé [12].*

Dans le cadre de l'apprentissage supervisé, différents types de classificateurs ont été développés pour atteindre une précision et une efficacité maximales. Chaque classificateur a ses avantages et ses inconvénients. Dans l'algorithme d'apprentissage supervisé existant, nous pouvons distinguer et regrouper de grandes familles [2] [12].

Dans ce qui suit, nous allons se focaliser sur les principaux algorithmes d'apprentissage supervisé.

2.6.2.1 Machine à vecteurs de support

Machine Vecteur Support(SVM) est une méthode de classification binaire par apprentissage supervisé, elle a été présentée par Vapnik en 1995. Cette méthode repose sur l'existence d'un classifieur linéaire dans un espace approprié. Puisqu'il s'agit d'un problème de classification à deux types, cette méthode fait appel à un jeu de données d'apprentissage pour apprendre les paramètres du modèle. Elle est basée sur l'utilisation de fonction dite noyau (kernel) qui permet une séparation optimale des données.

Le but de SVM est de trouver un classificateur pour séparer les données et maximiser la distance entre ces deux classes. Avec SVM, ce classificateur est un classificateur linéaire appelé hyperplan. Dans le schéma qui suit, on détermine un hyperplan qui sépare les deux ensembles de points.

Les points les plus proches, qui seuls sont utilisés pour la détermination de hyperplan, sont appelés vecteurs de support (voir Figure 2.5).

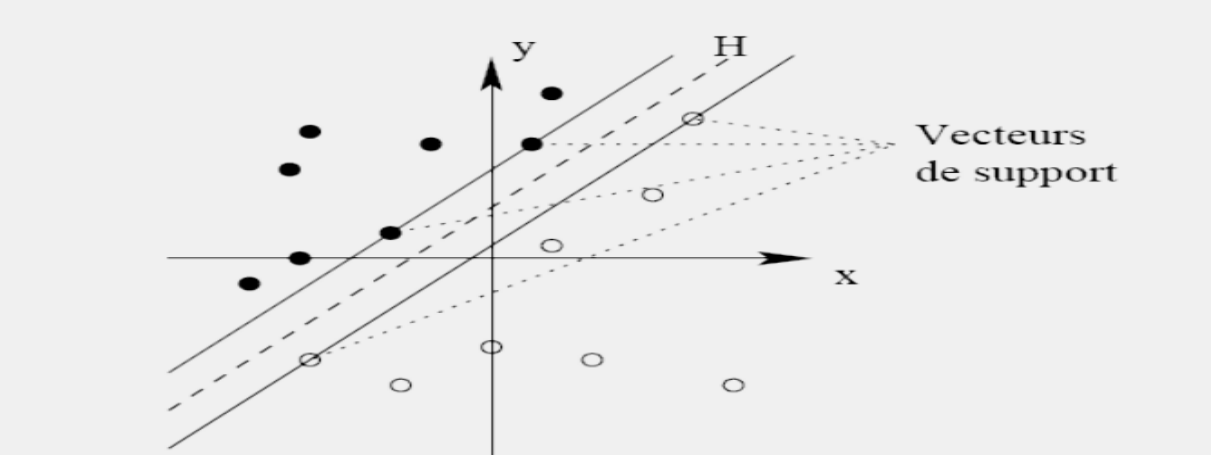

Figure 2.5 *Les vecteurs de support [2].*

2.6.2.2 Classification Naïve Bayésienne

La classification Naïve Bayésienne (NB) est un type de classification Bayésienne probabiliste simple basée principalement sur le théorème de Bayes avec une forte indépendance (dite naïve) des hypothèses. Elle met en œuvre un classifieur Naïve Bayésienne, ou classifieur naïf de Bayes, appartenant à la famille des classifieurs linéaires. Elle est utilisée dans plusieurs applications telles que la détection de courriels spam [17].

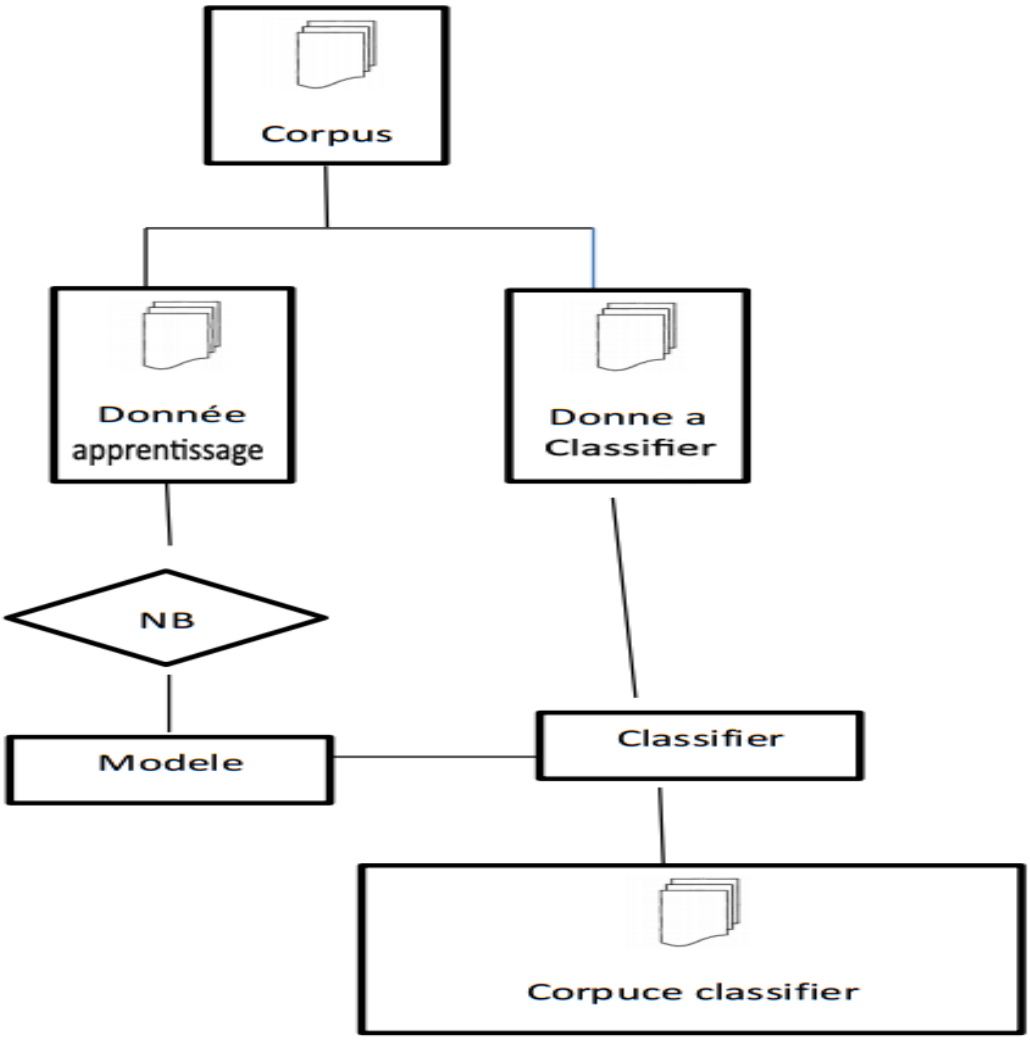

Figure 2.6 *Classification bayésienne [5].*

2.6.2.3 Méthode des k plus proches voisins

L'algorithme des k-plus proche voisin (K-ppv), traduction de k-nearest neighbors (KNN) en anglais, est une méthode d'apprentissage à base d'instances. Il n'a pas de phrase d'apprentissage. Enregistrer uniquement les documents dans l'ensemble d'études. Lorsqu'un nouveau document à classer arrive ; il est comparé au document d'apprentissage à l'aide d'une mesure de similarité. Ses k plus proches voisins sont alors considérés : on observe leur catégorie et celle qui revient le plus proche voisin est effectuée au document à classer. Cette méthode a prouvé son efficacité face au traitement des données textuel. La figure 2.7 présente le principe de K-ppv dans un espace à deux dimensions [18].

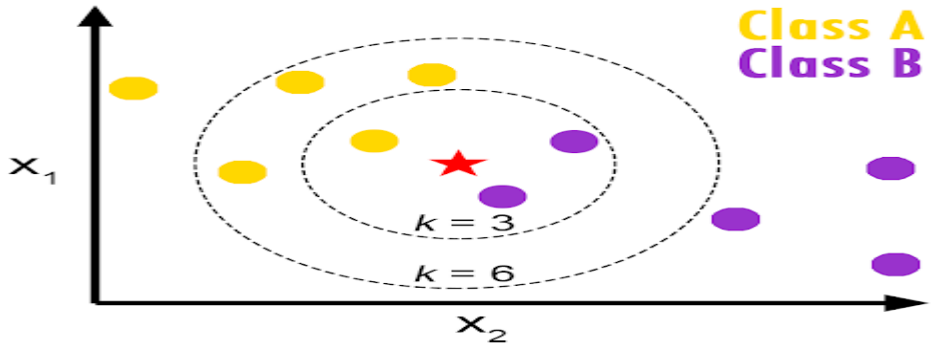

Figure 2.7 *K-ppv dans un espace à deux dimensions [26].*

Le choix de la valeur de K est dépendant de la taille de l'échantillon et des classes, et influence des résultats de classification. L'objet rond sera classifié triangle si k=3 et classifié carré si k=6.

2.6.2.4 Réseau de neurones

Un réseau de neurones artificiels est un système de technologie de l'information basé sur le fonctionnement du cerveau humain, les algorithmes dont la conception est à l'origine schématiquement inspirée du fonctionnement des neurones biologiques dont sont équipés les ordinateurs dotés de fonctions d'intelligence artificielle. La figure 2.8 explique le principe général d'une approche neuronale.

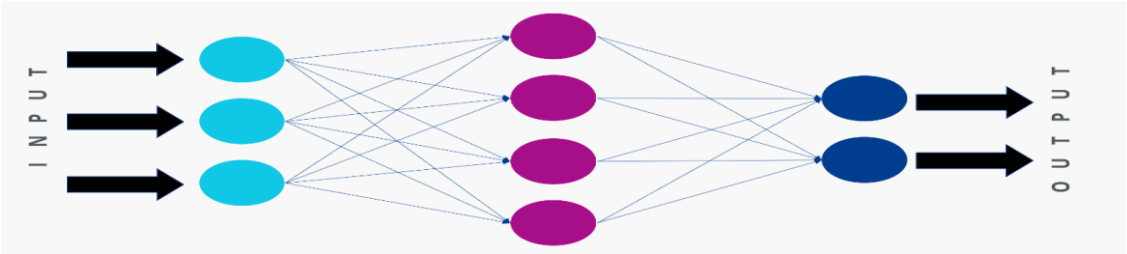

Figure 2.8 *Architecture générale d'un réseau de neurones artificiels [26].*

Un réseau de neurones artificiels peut être décrit comme un système composé d'au moins deux couches de neurones, la couche d'entrée et la couche de sortie comprenant généralement des « couches cachées ». Plus le problème à résoudre est complexe, plus le réseau de neurones artificiels ne doit comporter de couches. Chaque couche contient un grand nombre de neurones artificiels spécialisés.

2.6.2.5 Arbre de décision

Un arbre de décision est un outil d'aide à la décision représentant un ensemble de choix sous la forme graphique d'un arbre. Les différentes décisions possibles sont situées aux extrémités des branches (les « feuilles » de l'arbre), et sont atteintes en fonction de décisions prises à chaque étape.

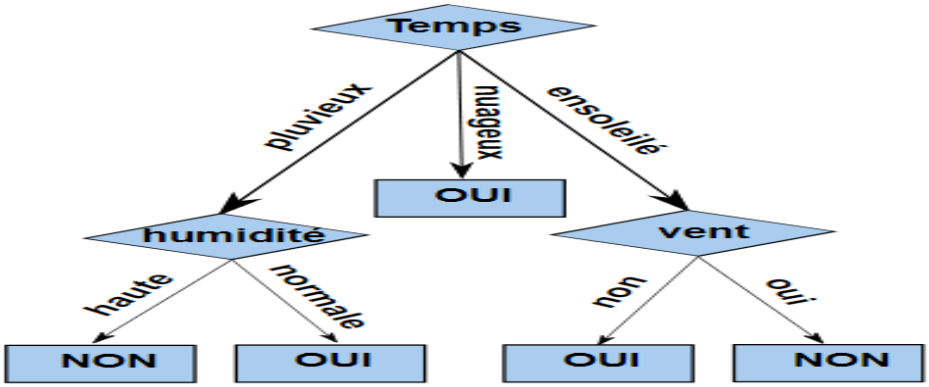

Figure 2.9 *Exemple d'arbre de décision [19].*

2.6.3 Domaines d'applications de la classification de textes

La classification de textes est utilisée dans de nombreuses applications. Parmi ces domaines on trouve :

- L'identification de la langue,
- La reconnaissance d'écrivains,
- La catégorisation de documents,
- Le filtrage, qui consiste à déterminer si un document est pertinent ou non, par exemple la détection des spams (les courriers indésirables).
- Le routage, qui permet d'affecter des documents à une ou plusieurs catégories en n, comme la diffusion sélective d'informations. Lors de la réception d'un document, l'outil choisira à qui l'envoyer en fonction de ses centres d'intérêt. Ces centres d'intérêt correspondent à des données personnelles [20].

2.7 Problèmes de classification de textes

Plusieurs difficultés peuvent s'opposer au processus de classification de textes, les principales sont les suivantes :

- **La redondance** : la redondance et la synonymie permettent d'exprimer le même concept par des expressions différentes, plusieurs façons d'exprimer la même chose. Contrairement aux données numériques, cette difficulté est liée à la nature du traitement des documents exprimée en langage naturel.
- **L'ambiguïté** : à la différence des données numériques, les données textuelles sont sémantiquement riches, du fait qui sont conçus et raisonnés par la pensée humaine. À cause de l'ambigüité, les mots sont parfois de mauvais descripteurs ; par exemple le mot

avocat peut désigner le fruit, le juriste, ou même au sens figuré, la personne qui défend une cause.

- **La graphie** : un terme peut comporter des fautes d'orthographe ou de frappe comme il peut s'écrire de plusieurs manières ou s'écrire avec une majuscule ou minuscule. Ce qui affectera sur la qualité des résultats. Parce que si un terme est orthographié de deux manières dans le même document (Ghelizane, Relizane), la simple recherche de ce terme avec une seule forme graphique abandonne la présence du même terme sous d'autres graphies.
- **Complexité de l'algorithme d'apprentissage :** un texte est représenté généralement sous forme de vecteur contenant les nombres d'apparitions des termes dans ce texte. Le nombre de textes qu'on va traiter est très important sans oublier le nombre de termes composant le même texte donc on peut bien imaginer la dimension du tableau (textes * termes) à traiter qui va compliquer considérablement la tâche de classification en diminuant la performance du système.
- **Présence-Absence de termes :** un mot dans le texte indique ce que l'auteur veut exprimer, il existe donc une relation implicite entre le mot et les concepts associés. Bien que l'on sache très bien, il y a plusieurs manières d'exprimer la même chose. L'absence d'un mot ne signifie pas nécessairement qu'aucun concept ne s'y rattache dans le document. Ce type de réflexion attentive nous oblige à utiliser avec soin les compétences d'étude en excluant des mots spécifiques.
- **Les mots composés :** les mots composés ne sont pas pris en charge, par exemple : Arc-en-ciel, Save-who-can, etc. Les nombres sont très grands dans toutes les langues, par exemple en réduisant 3 termes distincts pour traiter Arc-en-ciel. Cependant, les performances du système de classification sont assez impressionnantes. L'utilisation de la technologie n-gramme pour coder du texte peut grandement atténuer ce problème du mot composé [8].

2.8 Conclusion

Dans ce chapitre, nous avons abordé quelques étapes utilisées pour la classification des textes, et nous avons présenté le processus et les techniques de classification des textes, les principaux algorithmes d'apprentissage supervisé, et nous avons entamé par l'utilisation des algorithmes d'apprentissage dans le domaine de détection des spams. Enfin on peut dire que certains classificateurs fonctionnent mieux que d'autres, mais il est certain qu'il n'y a pas de classificateur parfait.

CHAPITRE 3
Filtre des spams Arabes

3.1 Introduction

Dans ce chapitre, nous proposons notre modèle de filtrage des spams basé sur le traitement automatique du langage naturel (NLP, Natural Language Processing). Ce modèle est testé sur plusieurs modèles d'apprentissage supervisé. Dans cette section, nous allons commencer par la présentation de quelques notions de base sur les spams Arabes, ensuite nous présentons notre problématique et notre proposition, puis la présentation de l'architecture de notre système, et nous expliquons comment construire et valider un modèle de détection avant sa mise en production, ensuite nous détaillons les méthodes d'évaluation des algorithmes, finalement on conclut notre chapitre.

3.2 Problématique

L'utilisation d'Internet dans le monde arabe connaît chaque jour une augmentation rapide. La population totale des pays arabes est d'environ 350 millions de personnes (5% de la population mondiale) et le total des utilisateurs d'Internet Arabes est d'environ 65 millions d'utilisateurs (3,3% du total des utilisateurs d'Internet) [21]. Dans ces dernières années, le rapport de spams par courrier électronique en arabe a beaucoup augmenté et les utilisateurs de courriers électroniques en arabe sont confrontés au problème du spam à très grande échelle mais la recherche dans ce domaine n'est pas aussi avancée que son homologue pour l'anglais pourriel.

La langue arabe se compose de 28 lettres écrites de droite à gauche et a une morphologie complexe. L'arabe expose deux genres : masculin et féminin, trois catégories de nombres : singulier, duel et pluriel. Alors que le singulier et le pluriel sont des catégories familières à la plupart des apprenants occidentaux, le dual est moins familier. Le dual en Arabe est utilisé à chaque fois que la catégorie de deux s'applique, où il se trouve dans les noms, les adjectifs, les pronoms ou les verbes. Les pluriels arabes sont divisés en deux catégories : régulier et cassé. Un nom a trois cas, le nominatif, l'accusatif et le génitif [24].

Le problème que nous voulons traiter est de créer un modèle capable de classer dynamiquement le spam arabe à partir de messages légitimes. Dans ce mémoire, un ensemble de données d'e-mails arabes est collecté et nous les avons utilisés pour entraîner différents algorithmes afin de les classer entre spam et non-spam. Nous avons opté pour *DecisionTreeClassifier* pour construire un modèle de détection de spam arabe. Des expériences

de validation croisée sont utilisées pour évaluer notre modèle. Dans notre cas l'algorithme *DecisionTreeClassifier* qui donne le meilleur résultat par rapport aux autres algorithmes.

3.3 Notre proposition

Après avoir détaillé certaines méthodes de détection de spam dans le chapitre précédent, nous avons introduit un modèle de détection de spam dans ce chapitre qui fournit des solutions à certaines des limitations observées à l'aide des fondements de l'apprentissage automatique. Notre méthode comprend l'analyse d'un ensemble de messages divisé en deux catégories, les messages de spam représentant des messages malveillants et les messages de ham représentant des messages bénins. L'objectif de cette analyse est de générer un modèle qui peut fournir de bonnes décisions de classification pour minimiser les taux d'erreur pour une communication plus fiable. Comme tous les modèles supervisés, l'apprentissage est basé sur des données anciennes, donc si vous pratiquez des mots que le modèle n'a pas encore reconnus, les performances de prédiction diminueront et des résultats incorrects seront estimés pour cela, nous avons considéré la relation qui relie chaque mot à un autre mot, qui est l'alphabet. Chaque mot est un ensemble de combinaisons ordonné, sauf des lettres répétées, pour former un sens unifié. Cette répétition de lettre est différente du courrier et du spam bénins, nous utilisons donc les deux méthodes le nombre de répétitions de chaque lettre et le nombre d'occurrences de chaque lettre.

3.4 Architecture du système

L'architecture de notre modèle de filtrage anti-spam est représentée dans la figure 3.1. Tout d'abord, en utilisons le corpus *Ensemble donnes* qui sont définis au chapitre 4. Le corpus de messagerie va passer par la phase de préparation ou prétraitement puis en utilisant l'un des algorithmes d'apprentissage supervisé pour construire un modèle qui permet de classer les nouveaux messages.

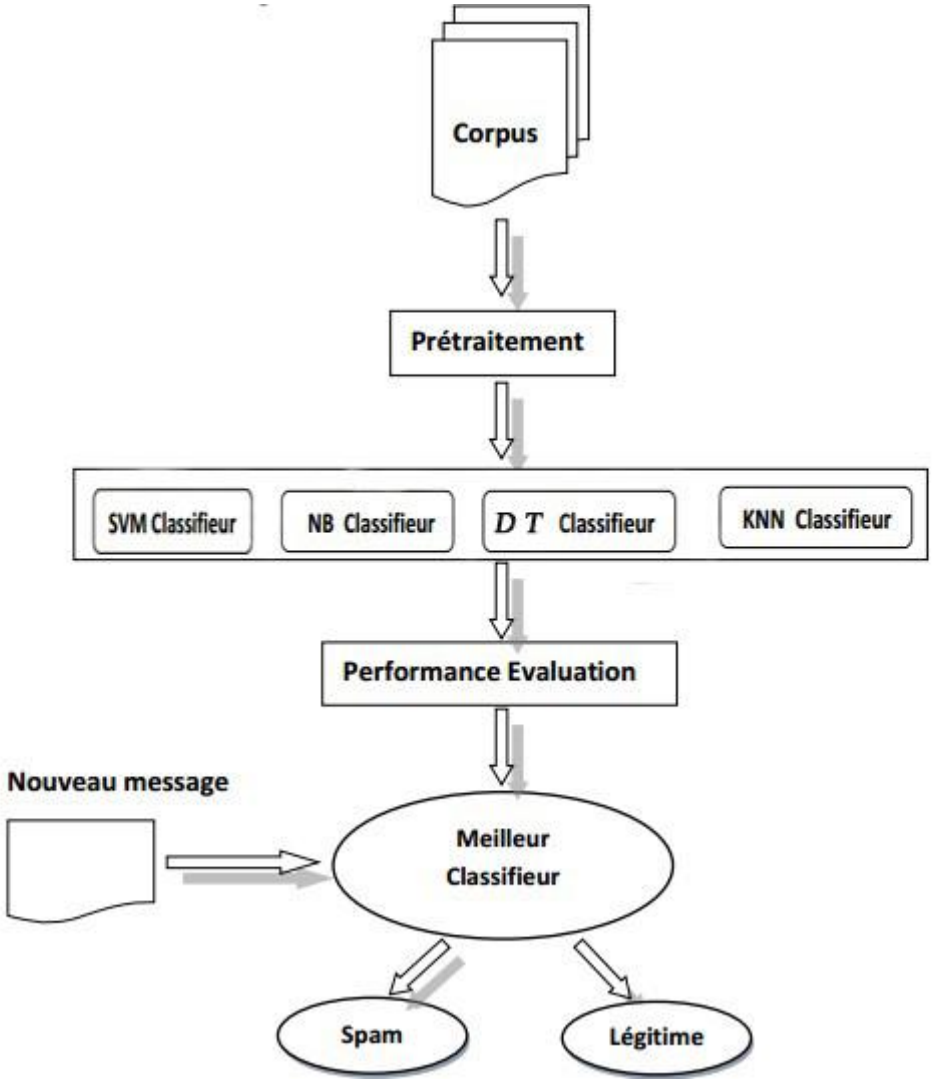

Figure 3.1 *L'architecture du modèle proposé.*

Les étapes d'apprentissage encerclées dans la figure 3.1 incluent différentes étapes de mise en œuvre de notre modèle avant utilisation. L'ordre d'exécution de ces étapes est représenté par des flèches. Après avoir créé le modèle, nous continuons à faire des prédictions. A chaque entrée d'un nouveau message, il sera filtré pour prédire que le message est un spam ou un ham.

3.5 Construction du modèle de détection

La première étape avant de construire un modèle de détection avec l'apprentissage automatique est de définir la cible, c'est-à-dire ce qu'on souhaite détecter. Pour créer un modèle de détection il faut :

- Recueillir des données d'apprentissage contenant des instances spam et non spam correspondant à la cible.
- Définir les attributs à extraire pour représenter les instances sous forme de vecteurs numériques.

- La construction d'un modèle de détection supervisé nécessite des données d'apprentissage pour lesquelles les labels, spam ou ham, sont connus. Ces données d'apprentissage doivent contenir des instances spam et non spam correspondant à la cible de détection.
- Choisir un type de modèle de classification adaptée aux contraintes opérationnelles.

3.6 Validation du modèle

La validation fait probablement partie des techniques les plus importantes utilisées par les scientifiques, car il est toujours nécessaire de valider la stabilité du modèle d'apprentissage automatique dans quelle mesure il se généraliserait à de nouvelles données. Nous devons être sûr que le modèle tient compte de la plupart des schémas des données et qu'il ne capte pas trop le bruit, ou en d'autres termes, son faible biais et variance.

3.7 Prétraitement des données

Le prétraitement est effectué pour supprimer les parties non pertinentes des données avant d'extraire une caractéristique. Le module de prétraitement se compose de cinq étapes consécutives : la tokenisation, la suppression du texte non arabe, la normalisation, la suppression des mots vides et la racinisation légère. Ces étapes, qui sont initialement effectuées sur le texte des critiques, sont importantes pour générer un texte prétraité prêt pour l'extraction et la classification des fonctionnalités.

- **Tokenisation :** divise le texte de l'examen en une séquence de jetons où chaque jeton représente un seul mot basé sur un caractère d'espacement.
- **Suppression de texte non arabe :** vérifie tous les jetons de la critique pour supprimer tout jeton non arabe de la critique.
- **Normalisation :** produit une forme cohérente à partir du texte saisi en convertissant les différentes formes du mot en une forme commune. Dans cette étape, les caractères du jeton de chaque avis sont vérifiés pour détecter s'ils sont dans leur forme normalisée ou non. Le tableau 3.1 montre la manière dont la normalisation du texte arabe est effectuée.

Lettres à remplacer	Remplacé par
ى ، ئ ، ئ	ي
، أ ، إ ، آ ، ٱ	ا
ة	ه
ؤ ، ۇ ، ۈ	و
,,,,,	Rien

Table 3.1 *Normalisation du texte Arabe [25].*

- **Suppression des mots vides :** supprime les mots dénués de sens qui apparaissent fréquemment dans le texte de la critique, ce qui peut améliorer le temps de réponse et réduire l'espace de l'index. Une liste de mots vides arabes contenant 700 mots vides est utilisée. Cette liste comprend des mots tels que (إلي ، من ، كان ، ، أو ، علي ، عن ، في ، كل ، أمام ، فقط قد لقد ،etc.).
- **Light Stemming :** renvoie le mot à sa forme originale. Pour les langues non arabes, un radical de base peut être soit préfixé, soit postfixé pour exprimer une syntaxe grammaticale. Cependant, en langue arabe, il est difficile de différencier certains mots arabes après leur radical, car certains mots ont la même racine alors qu'ils ont un sens complètement différent. Le tableau 2 montre un exemple de problème de racine arabe. En conséquence, la racine légère est utilisée pour éviter ce problème, ou un ensemble commun de préfixes et de suffixes est coupé d'un mot sans réduire un mot à sa racine.

mot arabe	Signification en français	Score de sentiment	Racine
لاعب	joueur	-1	ل
لعب	jouer	1	ل

Table 3.2 *Exemple de problème de radical Arabe.*

Préfixes et suffixes : pour ce faire, des listes de préfixes et de suffixes à une lettre, à deux lettres et à trois lettres sont établies. Le choix de ces listes est déterminé généralement selon des statistiques. Ces statistiques analysent les fréquences d'occurrence des préfixes et des suffixes sur les mots d'une grande collection de textes. La décision de tronquer un préfixe ou un suffixe d'un mot est faite selon de simples règles comme la longueur de mots. Par exemple, on ne peut pas tronquer un préfixe à trois lettres d'un mot de longueur quatre. Le tableau 3 présente une liste des préfixes et suffixes qui sont supprimés lors de l'application de la racine arabe légère.

Préfixes et suffixes supprimés	
Préfixes	وبال ، وكال، وفال ال ، وال ، بال فال ، لل ، كال ، ولل ،
Suffixes	ا ، ان ، ون ، ات ، ين ، يه ، وا

Table 3.3 *Liste des préfixes et suffixes [25].*

3.8 Méthode d'évaluation des algorithmes de classification

3.8.1 Mesure d'estimation de performance

Différentes métriques d'évaluation sont utilisées pour mesurer les performances des quatre méthodes de détection de spam Arabe. Ceux-ci le rappel (sensibilité), la spécificité, la précision et le score F1.

La précision : est la capacité d'une approche à différencier correctement le spam et les avis véridiques. Elle est mesurée en calculant la proportion de vrais positifs et de vrais négatifs dans tous les cas évalués.

Accuracy : $\left(\frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{TN}+\mathrm{FN}}\right) * 100\%$

Le rappel: mesure la proportion d'avis de spam correctement identifiés. Cela montre à quel point une approche est efficace pour détecter les critiques de spam.

Recall : $\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$

La spécificité : mesure la proportion d'avis véridiques correctement identifiés pour montrer à quel point une approche est efficace pour éviter les fausses alarmes.

Specificity : $\frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}}$

Le score F1: est la moyenne pondérée du rappel et de la précision.

F1 Score : $\frac{2\mathrm{TP}}{2\mathrm{TP}+\mathrm{FP}+\mathrm{FN}}$

où

Vrai Positif (TP) : nombre d'avis de spam correctement prédits, ce qui signifie que la valeur de la classe réelle est du spam et que la valeur de la classe prédite est également du spam.

Vrai Négatif (TN) : nombre d'avis véridiques correctement prédits, ce qui signifie que la valeur de la classe réelle est véridique et que la valeur de la classe prédite est également véridique.

Faux Positif (FP) : nombre d'avis de spam mal prédits, ce qui signifie que la valeur de la classe réelle est du spam mais que la valeur de la classe prédite est véridique.

False Negative (FN) : nombre d'avis véridiques prédits de manière incorrecte, ce qui signifie que la valeur de la classe réelle est véridique mais que la valeur de la classe prédite est du spam.

Le modèle de détection n'est pas parfait et il peut produire des erreurs de prédiction. Avant sa mise en production, il doit être évalué, c'est-à-dire que la pertinence des alertes

générées est mesurée. L'estimateur de performance le plus connu est le taux de mauvaise classification, qui est égal au pourcentage d'instances mal classées. Cependant, en cas de détection de spam les données sont généralement très asymétriques (la proportion d'instances malveillantes est faible), et le taux d'erreur ne peut pas estimer correctement les performances du classificateur dans cette situation.

Voici un exemple présentant les limites du taux d'erreur de classification : on considère 100 instances : 2 spam et 98 ham. Dans cette situation, un modèle de détection niais prédisant toujours ham aura un taux d'erreur de classification seulement 2% alors qu'il n'est pas capable de détecter la moindre instance malveillante.

3.8.2 Matrice de confusion

Afin d'analyser correctement les performances du modèle de détection, la première étape inclut l'écriture d'une matrice de confusion, qui prend en compte deux types d'erreurs possibles : Faux positifs, c'est-à-dire faux positifs causés par des cas bénins et faux négatif, c'est-à-dire une instance malveillante non détectée. La figure 3.2 décrit une matrice de confusion [26].

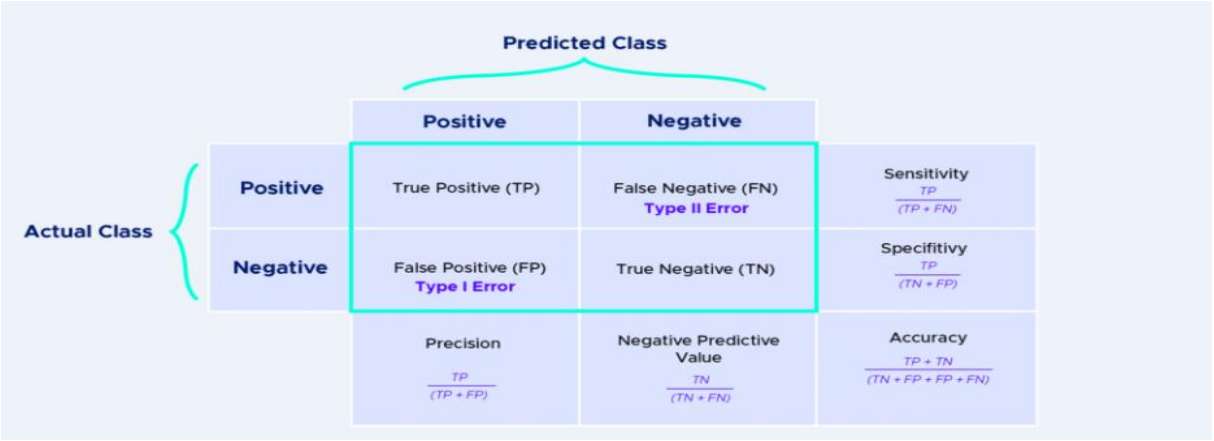

Figure 3.2 *Matrice de confusion [26].*

La matrice de confusion permet d'exprimer des indicateurs de performance tels que le taux de détection ou le taux de faux positifs :

Taux de faux positifs : $\frac{F\,P}{F\,P\,+V\,N}$

Taux de détection : $\frac{VP}{V\,P\,+F\,N}$

Un modèle de détection de spam doit être évalué avec ces deux estimateurs de performance pris ensemble. En effet, le taux de faux positifs doit être faible pour que l'opérateur de détection ne soit pas submergé par les fausses alertes. Le taux de détection doit quant à lui

être élevé pour éviter que trop de spams restent non détectées. Le seuil de détection détermine la sensibilité de la détection : baisser ce seuil augmente le taux de détection, mais aussi le taux de fausses alertes. Il est ainsi fixé par l'administrateur du système de détection en fonction du compromis souhaité entre taux de détection et taux de faux positifs. Les estimateurs de performance que nous venons de présenter dépendent de la valeur du seuil de détection.

3.8.3 La courbe de ROC

La courbe de ROC nous permet d'estimer les performances, qui a l'avantage d'être indépendant de ce seuil, est souvent utilisé en détection : la fonction d'efficacité du récepteur, plus fréquemment désignée sous le terme de courbe ROC. Cette courbe représente le taux de détection en fonction du taux de faux positifs pour divers seuils de détection. Pour un seuil de 100%, les taux de détection et de fausses alertes sont nuls, et pour un seuil de 0% ils sont tous deux à 100%. Un modèle de détection est d'autant plus performant que sa courbe est proche du coin supérieur gauche : un fort taux de détection pour un faible taux de fausses alertes. L'aire sous la courbe ROC, appelée AUC (Area Under the ROC Curve), est souvent calculée pour estimer la performance d'un modèle de détection indépendamment du seuil de détection, et sa valeur doit être proche de 1 [23] [34]. Un classifieur prédisant de manière aléatoire la probabilité de malveillance a pour courbe ROC la droite rouge représentée sur la figure 3.3. Ainsi, la courbe ROC d'un classifieur doit toujours être au-dessus de cette droite (sinon un classifieur aléatoire a de meilleures performances...), et l'AUC est au minimum de 0.5. La courbe ROC est non seulement un estimateur de performance, mais elle permet aussi à l'administrateur de choisir la valeur du seuil de détection en fonction du taux de détection souhaité ou du taux de fausses alertes toléré.

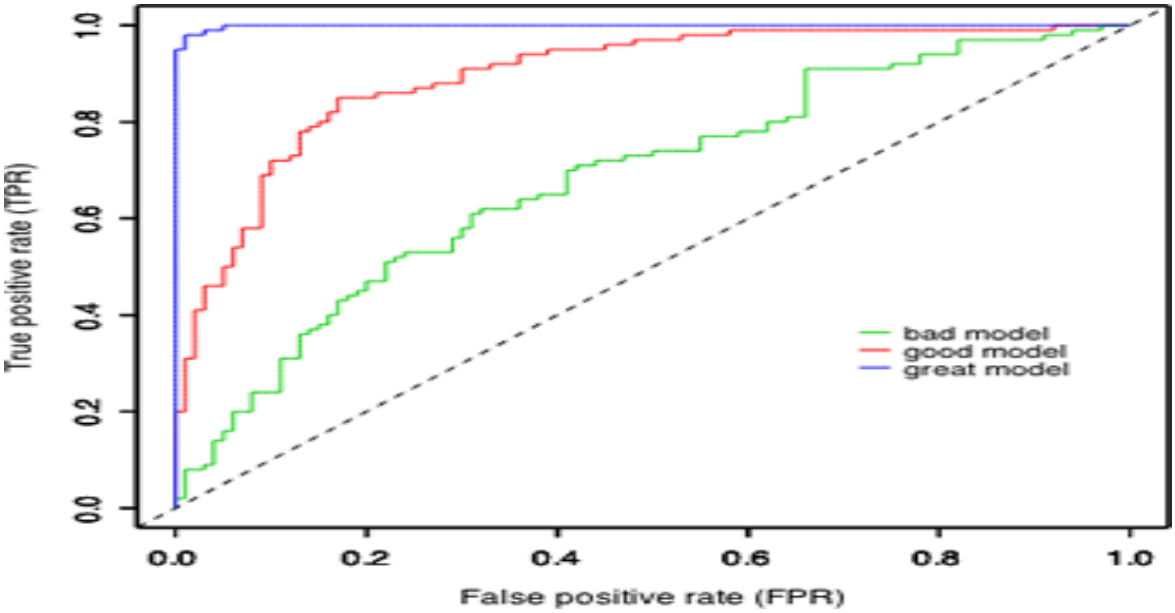

Figure 3.3 *La courbe ROC [23].*

3.9 Conclusion

Dans ce chapitre, nous avons présenté la problématique, et de notre système de filtrage des spams, et nous avons présenté l'architecture de notre système de filtrage des spams, et les méthodes d'évaluation des algorithmes de classification, basée sur machine learning. Afin de tester une nouvelle approche de description des résultats de la classification supervisée en langue Arabe.

CHAPITRE 4
Implémentation

4.1 Introduction

Après avoir donné toute la théorie que nous voyons nécessaire pour la réalisation de notre modèle. Il est temps de mettre en œuvre notre modèle qui permet de filtrer les messages spam et non spam. Dans cette section nous allons présenter les outils et l'environnement de programmation utilisés pour la réalisation de l'étude proposée, ainsi les bibliothèques essentielles pour l'apprentissage automatique, finalement nous allons mettre en clair les étapes de l'implémentation montrée dans l'architecture du modèle proposé, on conclut notre chapitre.

4.2 Outils et environnement de réalisation

Dans ce qui suit, nous définirons l'environnement de développement python, Anaconda, Spyder ainsi que Jupyter qui ont servi d'outils au développement de notre modèle.

4.2.1 Python :

Python est un langage de programmation (au même titre que le C, C++, fortran, java, etc.), le plus utilisé dans le domaine du Machine Learning, du Big Data et de la Data Science, développé en 1989. Ses principales caractéristiques sont les suivantes :

– « Open-source » : son utilisation est gratuite et les fichiers sources sont disponibles et modifiables.

– Importante quantité de bibliothèques disponible pour le calcul scientifique, les statistiques, les bases de données, etc.

– Grande portabilité : indépendant vis à vis du système d'exploitation (linux, windows, Mac OS).

– Orienté objet.

–Typage dynamique : le typage (association à une variable de son type et allocation zone mémoire en conséquence) est fait automatiquement lors de l'exécution du programme, ce qui permet une grande flexibilité et rapidité de programmation, mais qui se paye par une surconsommation de mémoire et une perte de performance.

– Présente un support pour l'intégration d'autres langages [37].

4.2.2 Anaconda :

Anaconda est un gestionnaire de paquets, un gestionnaire d'environnement, une distribution Python / R de science des données et une collection de plus de 1500 packages open source. Anaconda est gratuit, facile à installer et offre un support gratuit à la communauté [38].

4.2.3 Jupyter Netbook :

Jupyter Notebook est une application Web à source ouvert qui vous permet de créer et de partager des documents contenant du code en direct, des équations, des visualisations et du texte narratif. Les utilisations incluent : nettoyage et transformation de données, simulation numérique, modélisation statistique, visualisation de données, apprentissage automatique, etc.

4.2.4 Spyder :

Spyder est un environnement scientifique puissant écrit en python, pour python, et conçu par et pour les scientifiques, les ingénieurs et les analystes de données. Il offre une combinaison unique des fonctionnalités avancées d'édition, d'analyse, de débogage et de profilage d'un outil de développement complet avec l'exploration des données, l'exécution interactive, l'inspiration approfondie et les superbes capacités de visualisation d'un progiciel scientifique[39].

4.2.5 Bibliothèques essentielles pour l'apprentissage automatique

- **Scikit-learn :** Est l'une des bibliothèques de machine learning les plus populaires. Il prend en charge de nombreux algorithmes d'apprentissage supervisé et non supervisé. Les exemples incluent les régressions linéaires et logistiques, les arbres de décision, le regroupement, k moyennes, etc. [40].
- **Pandas :** Est une bibliothèque très populaire qui fournit des structures de données de haut niveau, simples à utiliser et intuitives. Elle possède de nombreuses méthodes intégrées pour le regroupement, la combinaison de données et le filtrage, etc. [41].
- **Matplotlib :** Il s'agit d'une bibliothèque Python standard utilisée par tous les scientifiques pour la création de graphiques et de tracés 2D. C'est assez bas niveau, ce qui signifie qu'il nécessite plus de commandes pour générer des graphiques et des figures plus jolis que dans certaines bibliothèques avancées [42].
- **NumPy** : Est un package de traitement de matrice à usage général. Il fournit un objet de tableaux multidimensionnels hauts performances et des outils pour travailler avec ces tableaux. C'est le paquet fondamental pour l'informatique scientifique avec Python. Il

contient diverses fonctionnalités, notamment les plus importantes :

– Un puissant objet de tableau à N dimensions.

– Fonctions sophistiquées (diffusion).

– Outils d'intégration du code C / C ++.

– Algèbre linéaire, transformée de Fourier et capacités de nombres aléatoires [43].

4.3 Expérimentation, évaluation et discussion

Cette section comprend les résultats expérimentaux et l'évaluation de l'architecture proposée dans le chapitre précédent, y compris les quatre méthodes de détection de spam Arabe mentionnées ci-dessus.

4.3.1 Création d'un ensemble de données

Après une longue recherche nous n'avons pas pu trouver de données contenant du texte brute en langue Arabe. Pour cela, nous avons opté pour la solution de créer un mini-dataset qui contient un total de 208 messages, dont 61 sont étiquetés comme des spams et 146 qui sont des messages non spam. Le dataset est sous forme d'un fichier (.xlsx : ensemble donnes) qui est un fichier excel. Comme illustre la figure 4.1.

```
Entrée [182]: #importation des donnes
              data=pd.read_excel('enssemble donnees.xlsx')
              data.head(5)
```

Out[182]:

	class	message
0	spam	... عزيز العميل تم حظر بطاقة الصراف الآلي الخاصة
1	spam	...عزيزي العميل نرجو تحريث بياناتك البنكية عدم تح
2	spam	...عزيزي العميل الصراف الآلي الخاص بك محظور فا تص
3	spam	...موقع عجيب تسوى منه سيرتك الذ Novo Resumes موقع
4	spam	...لا تفوتها! احصل على 250 دقيقة محلية من موبايلي

Figure 4.1 *Premières lignes de dataset.*

4.4 Organisation et reformulation du dataset

Une mauvaise organisation du dataset invoque l'anarchie des données et exploite plus d'espace mémoire avec une détermination critiquable. L'organisation comprend :

1. Nettoyage de données à savoir suppression des valeurs manquantes.
2. Nous avons identifié la variable cible qu'est la variable class nous l'avons transformée en variable numérique du type float.
3. Structurer et transformer les données numériques exploitables par les algorithmes d'apprentissage automatique à l'aide des techniques de (NLP).

4.4.1 Nettoyage de données

Notre dataset contient des colonnes qui contient des valeurs vides nous avons 2 valeurs dans la colonne message qui sont nuls, nous devant faire le nettoyage, on va éliminer les valeurs manquantes la figure 4.2 montres toutes les colonnes ainsi celle qui contient des valeurs vides, et la figure 4.3 montres comment supprimer ces colonnes.

```
Entrée [61]: #le nombres des valeurs manquante dans chaque colonne
             data.isnull().sum()

Out[61]: class      0
         message    2
         dtype: int64
```

Figure 4.2 *Affichage des colonnes et nombre des valeurs manquant.*

```
Entrée [62]: # la selectionne des lignes qui contient des valeurs vides
             index_with_nan = data.index[data.isnull().any(axis=1)]
             index_with_nan.shape
             #La suppression des lignes qui contient des valeurs vides
             data.drop(index_with_nan,0, inplace=True)
```

Figure 4.3 *Suppression des lignes vides.*

4.4.2 Transformation des données

La colonne class contient des valeurs string nous devont les transformer aux valeurs numériques ont remplaçant spam par 1 et ham par 0. La figure 4.4 montre notre dataset après la transformation de la colonne class.

```
Entrée [60]: #remplacement spam par 1 et ham par 0
             data['class']=data['class'].map({'ham':0, 'spam':1})
             data.head(5)
```

Out[60]:

	class	message
0	1	... عزيز العميل تم حظر بطاقة الصراف الآلي الخاصة
1	1	...عزيزي العميل نرجو تحريث بياناتك البنكية عدم تح
2	1	...عزيزي العميل الصراف الآلي الخاص بك محظور فا ئص
3	1	...موقع عجيب تسوى منه سيرتك الذ Novo Resumes موقع
4	1	...لا تفوتها! احصل على 250 دقيقة محلية من موبايلي

Figure 4.4 *Transformation des données.*

4.4.3 Standardisation des données

Avant de parler sur la normalisation, il est très important pour nous d'en connaître la nécessité. Et nous voyons, les ensembles de données que nous utilisons pour construire un modèle pour un énoncé de problème particulier sont généralement construits à partir de diverses sources. Ainsi, on peut supposer que l'ensemble de données contient des variables/caractéristiques d'échelles différentes. Pour que notre modèle d'apprentissage

automatique fonctionne bien, il est très nécessaire que les données aient la même échelle en termes de fonctionnalité pour éviter les biais dans le résultat. Par la suite, la mise à l'échelle des caractéristiques est considérée comme une étape importante avant la modélisation.

```
Entrée [17]:
#standardisation (met les donnes en type accessible pour trainner et tester)
scaler = StandardScaler()
df_scaler = scaler.fit_transform(x)
df_scaler

Out[17]: array([[-0.13564414, -0.17236639, -0.02500146, ..., -0.08999732,
                 -0.0767578 , -0.40806327],
                [-0.13564414, -0.09861572,  0.03667328, ..., -0.08999732,
                 -0.0767578 , -0.39287944],
                [-0.13564414, -0.12543414,  0.00174743, ..., -0.07454286,
                 -0.0767578 , -0.39287944],
                ...,
                [-0.51622758, -0.39864687, -0.13525355, ..., -0.21363301,
                 -0.0767578 ,  0.29039312],
                [-0.36822291, -0.5846997 ,  0.11149504, ..., -0.2599964 ,
                 -0.0767578 ,  0.5029668 ],
                [-0.6219452 , -0.56290973,  0.30065966, ..., -0.2599964 ,
                 -0.0767578 ,  0.18410627]])
```

Figure 4.5 *Standardisation des données.*

4.5 Création du modèle :

Après avoir préparé l'ensemble des données, on passe maintenant à l'appel des algorithmes. La première étape est la décomposition du jeu de données en deux parties : une partie réservée pour l'ensemble du test et l'autre partie restante est pour l'apprentissage et cela est géré d'une façon aléatoirement dont on spécifie grâce au module de sklearn_model_selection du répertoire train_test_split. La première partie des données est comme suit : 80% pour l'apprentissage et les 20% restants sont pour le test, comme la montre la figure ci-dessous :

```
Entrée [19]: #20% pour le test et 80% pour apprentissage
x_train, x_test,y_train, y_test=train_test_split(x,y, test_size=0.2,  random_state=42)
```

Figure 4.6 *Décomposition du jeu de données.*

4.6 Evaluation du modèle

Une fois notre modèle est paramétré et entrainé sur un jeu de validation, là on arrive à l'étape de l'affichage de la précision et l'erreur quadratique des modèles. Ces modèles entrainés renvoi les résultats représentés sur la figure 4.7.

Out[23]:

	name	Accuracy Score	erreur
0	MultinomialNB	0.928571	0.267261
1	LogisticRegression	0.928571	0.267261
2	KNeighborsClassifier	0.809524	0.436436
3	DecisionTreeClassifier	0.952381	0.218218
4	RandomForestClassifier	0.928571	0.267261
5	GradientBoostingClassifier	0.928571	0.267261
6	SVC	0.904762	0.308607

Figure 4.7 *Résultat de l'évaluation des performances des algorithmes.*

Nous constatons que le meilleur algorithme qui donne de bons résultats c'est celui qui donne la précision élevée et l'erreur moins dans notre cas nous avons le modèle ***DecisionTreeClassifier*** est le meilleur, on a 0.952 sur 1 ce qu'il veut dire *95.23%* pourcent sur *100%*, et 0.218 de l'erreur quadratique qui est proche de 0. La figure 4.8 et 4.9 représente les diagrammes à barres des modèles scores et l'erreur.

```
Entrée [45]: plt.figure(figsize = (20,10))
             sns.barplot(x = "name", y = "Accuracy Score", data=df0)
             plt.title("Model Comparaison vs Score")

Out[45]: Text(0.5, 1.0, 'Model Comparaison vs Score')
```

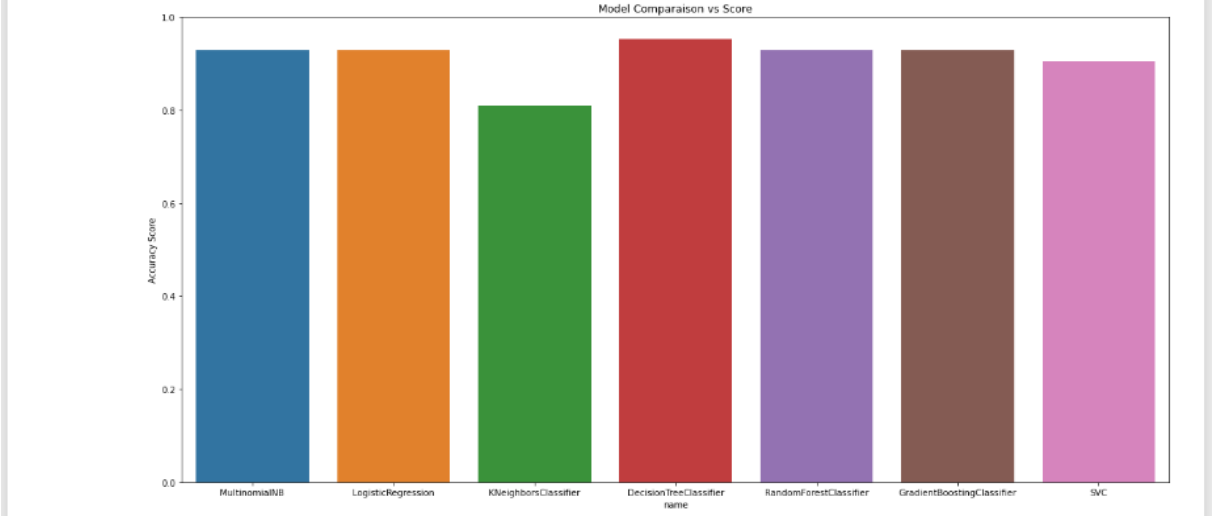

Figure 4.8 *Résultat de comparaison du score.*

```
Entrée [46]: plt.figure(figsize = (20,10))
             sns.barplot(x = "name", y = "erreur", data=df0)
             plt.title("Model comparaison vs erreur")

Out[46]: Text(0.5, 1.0, 'Model comparaison vs erreur')
```

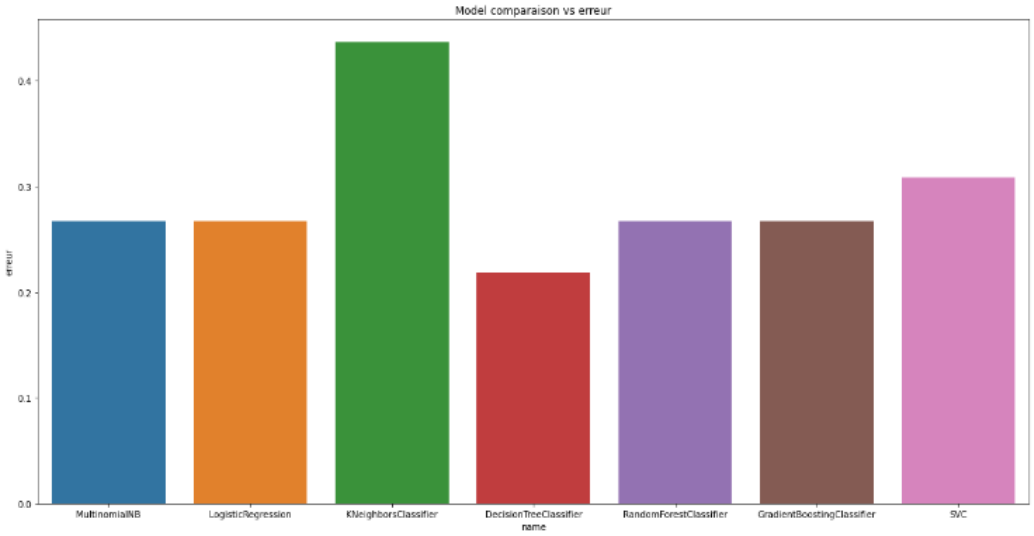

Figure 4.9 *Résultat de comparaison d'erreur.*

4.6.1 Matrice de confusion

La matrice de confusion permet d'exprimer des indicateurs de performance tels que le taux de détection ou le taux de faux positifs. Un modèle de détection doit être évalué avec ces deux estimateurs de performance pris ensemble. En effet, le taux de faux positifs et le taux faux négatif doivent être faible pour que l'opérateur de sécurité supervisant le système de détection ne soit pas submergé par les fausses alertes, la figure 4.10 montre le résultat obtenu de meilleur modèle.

```
------------------------------
DecisionTreeClassifier :
la precision est: 0.9523809523809523
l'erreur quadratique moyenne est: 0.21821789023599236
la matrice est:
 [[33  0]
 [ 2  7]]
------------------------------
```

Figure 4.10 *Le résultat de la matrice de confusion.*

4.6.2 Courbe de roc

La courbe illustrée dans la figure 4.11 nous permettent de visualiser les probabilités pour les mesures, taux de faux positifs en fonction du vrai positif.

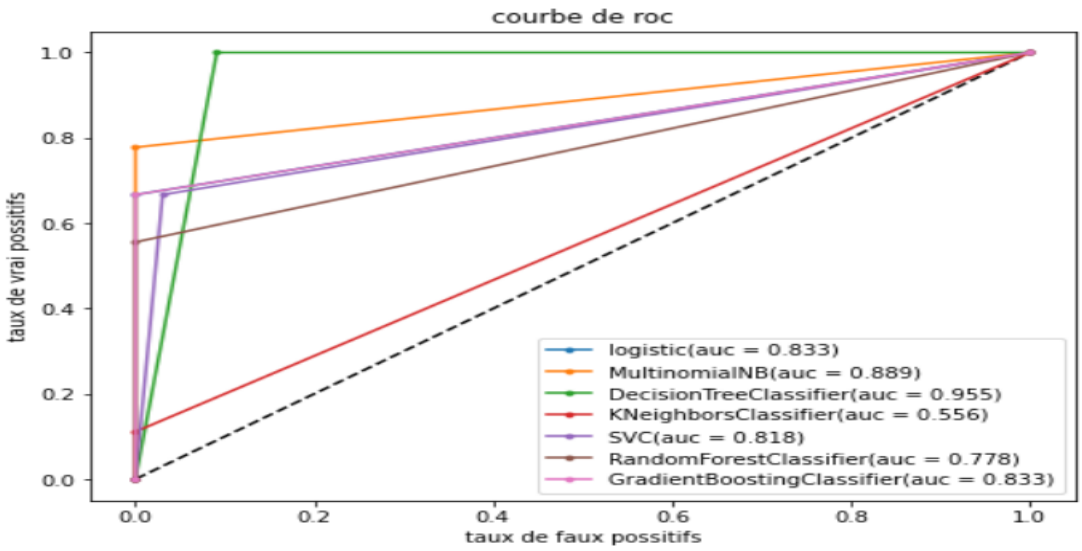

Figure 4.11 *Courbe ROC*

4.7 Interfaces Graphiques de l'application web

Nous avons créé cette page sous forme d'application web. On peut saisir quelques messages puis en cliquant sur le bouton filtré, le résultat de prédiction s'affiche dans la fenêtre en bas, la figure 4.12 et 4.13 montre l'interface web de notre modèle.

Figure 4.12 *Interface web du modèle proposé (spam).*

Figure 4.13 *Interface web du modèle proposé (non spam).*

4.8 Conclusion

Dans ce chapitre, nous avons expliqué les étapes de l'implémentation du modèle et les différents outils nécessaires pour le réaliser. On a étudié les résultats obtenus dans les différents algorithmes SVM, NB et KNN, RF, DT avec des différentes représentations et nous avons opté de prendre l'algorithme ***DecisionTreeClassifier*** comme meilleur classifieur grâce à ces bons résultats.

Conclusion générale

Le domaine de détection de spam a particulièrement progressé ces dix dernières années, grâce à l'introduction des techniques héritées de l'apprentissage automatique qui ont amélioré le taux du filtrage de spam, par la progression de classification des emails en spam et légitime.

La langue Arabe est un moyen de communication à travers le support informatique qui a été longtemps appréhendée avec beaucoup d'hésitation par la communauté scientifique, notamment celle du monde arabe où cet outil trouvera beaucoup d'utilisations importantes. En fait, la langue arabe et ses difficultés liées, notamment le problème de l'ambiguïté issue de l'absence des voyelles, et le problème d'absence de travaux publiés sur l'extraction de l'information en langue Arabe, tout cela pose un énorme défi difficile à surmonter. Malgré tout cela, nous avons osé nous aventurer dans ce domaine et on peut dire que, vu les résultats obtenus, nous pensons qu'on a quand même pu relever ce défi et par la même occasion apprendre beaucoup de nouvelles connaissances tout au long de la réalisation de ce travail tel que le traitement automatique du langage naturel (notamment, la langue Arabe).

L'objectif de notre travail se dirigeait vers le développement d'une approche d'apprentissage automatique, qui consiste à construire un modèle qui détecte les Émail bénignes et spam, afin d'améliorer la performance du système de filtrage avec DecisionTreeClassifier. Malgré les performances de ce système, il est intéressant de continuer le travail sur d'autres corpus plus riches en terme de donnés, et appliquer une combinaison avec d'autres classifieurs tels que : naive bayes, KNN, RF, Log Reg, etc.

Comme perspectives nous envisageons d'ajouter d'autres langues pour rendre le système multi-langues, et construire un modèle qui détecte les URLS bénignes et malveillantes. Et nous comptons faire une étude sur les messages contenant des images.

Références Bibliographiques

[1] P. Guillon, Etat de l'art du spam, solutions et recommandations, Mémoire de master, École de Gestion de Genève (HEG-GE), 2008.

[2] B. Naouel, Détection de courriels indésirables par apprentissage automatique, Mémoire de magister, Université Oran, 2012.

[3] F. Barigou, et B. Atmani, Voting Multiple Classifiers Détections for spm detection, Internation Conference on Information Technology and e-Service. ICTeS'2012. Sousse, Tunisia, 2012.

[4] P. Trudel, F. Abran, G. Dupuis, Analyse du cadre réglementaire Québécoise et étranger à l'égard du pourriel, de l'hamerçonnage des logiciels espions, Paris, 2017.

[5] B. Hassan, Algorithme de boosting et méta-heuristique basée sur la PSO pour la détection et le filtrage de spam, thèse de master, Université Tahar Moulay-SAIDA, 2013.

[6] G. Schryen, Anti-Spam Measures Analysis and Design, Berlin Heidel berg New York, Springer, 2010.

[7] T. Bayes, 1763, An Essay towards solving a Problem in the Doctrine of chances, Philosophical Transactions of the Royal Society of London, Vol. 53.

[8] N. Rimoucheet H. Hachemi, Amélioration du produit scalaire via les mesures de similarités sémantiques dans le cadre de la catégorisation des textes, Mémoire de master, Université Abou Bakr Belkaid– Tlemcen, 2015.

[9] R. Jalam, Apprentissage automatique et catégorisation de textes multilingues, Thèse de doctorat, Université Lumière Lyon 2, France, Juin 2003.

[10] S. Ouali et A. Chekaiem, L'Extraction de Mots Pertinents pour la Classification de Textes Arabes, Mémoire de Master, Université Ahmed Draia - Adrar, 2019.

[11] M. S. El Bazzi, T. Zaki, D. Mammass, A. Ennaji, Indexation automatique des textes arabes : état de l'art, Mémoire de master, 2016.

[12] K. BOUKHARI, Un Nouvel Algorithme de Stemmatisation pour l'Indexation Automatique de documents non-structurés : Stemmer SAID, Mémoire de master, Université de Monastir - Tunisie, 2013.

[13] R. Jalam, and J.H. Chauchat, Pourquoi les n-grammes permettent de classer des textes? Recherche de mots-clefs pertinents à l'aide des n-grammes caractéristiques.

[14] Furnkranz Johannes, Tom Mitchell, Ellen Rilo, A Case Study in Using Linguistic Phrases for Text Categorization on the WWW School of Computer Science, Mémoire de master, Carnegie Mellon University.

[15] A.GERON, Machine Learning Avec Scikit-Learn, Dunod, Paris, 2017.

[16] A. Burkov, Machine Learning Engineering - Draft, pp. 12-274.

[17] F. Ait Mahammed, Approches d'apprentissage automatique pour la détection du spam web, Mémoire de magister, Univ-MONTRÉAL, 2018.

[18] B. Fatiha, Contribution À la catégorisions de textes et À l'extraction d'information, mémoire de doctorat, Université Oran, 2012.

[19] K. ABIDI, « La catégorisation de texte Multilingue », Mémoire de Magister, Ecole supérieur d'Informatique, Algérie, 2010-2011.

[20] G. Charaf Eddine, Détection des spams se basant sur les techniques de classification, Mémoire de master, Univ-MOHAMED BOUDIAF - M'SILA, 2018.

[21] A. H. Wahbeh and M. Al-kabi, "Comparative Assessment of the Performance of Three WEKA Text Classifiers Applied to Arabic Text," vol. 21, no. 1, pp. 15–28, 2012.

[22] J. Hovold, Naive Bayes Spam Filtering Using Word-Position-Based Attributes.

[23] M. Nassou et D. SAADI., Systèmes de détection d'intrusions et machine learning, mémoire de master, Université A/Mira de Béjaïa, 2020.

[24] K. C. Ryding, A Reference Grammar of Modern Standard Arabic, 1998.

[25] I. Abu El-Khair, 2006. Effects of Stop Words Elimination for Arabic Information Retrieval: A Comparative Study. International Journal of Computing & Information Sciences, Volume 4, Number 5, pp. 119 – 133.

[26] M. FEREIRRA, Malicious URL Detection using Machine Learning Algorithms,Lusofona, Mémoire de master, University of Porto, Portugal, 2019. pp. 1-21.

[27] K. Adi, L. Hamza, and L. Pene, Automatic security policy enforcement in computer systems. Computers & Security. 73, pp. 156-171. (2018).

[28] K. Adi, L. Hamza et L. Pene. Formal modeling for security behavior analysis of computer systems. In Proceedings of the 2008 International MCETECH Conference on eTechnologies, IEEE, pp. 49-59, Montreal, Quebec, Canada, (2008).

[29] L. Hamza, K. Adi, and K. El ghemhioui. Automatic generation of attack scenarios for intrusion detection systems. In International Conference on Internet and Web Applications and Services, IEEE, pp. 205-211, Guadeloupe, France, (2006).

[30] L. Hamza, K. Adi, Formal Technique for Discovring Complex attacks in computer systems. In Proceedings of the 2007 conference on New Trends in Software Methodologies, Tools and Techniques, IOS Press, pp. 185-199, Rome, Italie, (2007).

[31] L. Hamza, Génération automatique de scénario d'attaques pour les systèmes de détection, mémoire de magistère, Université de Béjaia-Abderrahmane Mira, (2005).

[32] L. Hamza, Intruder Model for Generating Attack Scenarios in Computer Systems. International Journal of Information and Computer Security.13, pp. 428-443. (2020).

[33] L. Hamza, Modèle d'intrus pour générer des attaques complexes dans les systèmes informatiques. In Proceedings of the 2018 International Symposium ISKO-Maghreb, pp. 81-86, Bejaia, Algérie, (2018).

[34] L. Hamza, Modèle d'intrus pour générer des attaques complexes dans les systèmes informatiques. In Proceedings of the 2018 International Symposium ISKO-Maghreb, pp. 81-86, Bejaia, Algérie, (2018).

Références Webliographiques

[35] www.altospam.com/actualite/2019/05/statistiques-sur-les-spams, Consulté le 17/02/2021.

[36] www.definitions-marketing.com/definition/filtrage-anti-spam/, Consulté le 20/02/2021.

[37] https ://docs.python.org/, Consulté le 9/07/2021.

[38] https ://docs.anaconda.com/anaconda/, Consulté le 09/07/2021.

[39] https ://docs.spyder-ide.org/current/index.html, Consulté le 08/07/2021.

[40] https ://www.kite.com/python/docs/sklearn, Consulté le 08/07/2021.

[41] https ://pandas.pydata.org/docs/, Consulté le 08/07/2021.

[42] https ://matplotlib.org/, Consulté le 08/07/2021.

[43] https ://numpy.org/doc/stable/user/whatisnumpy.html, Consulté le 09/07/2021.

RÉSUMÉ

Le courrier électronique rend vraiment service aux usagers, c'est un moyen rapide et économique pour échanger des informations. Cependant, les utilisateurs se retrouvent assez vite submergés de quantités de messages indésirables appelés aussi spam. Le spam est rapidement devenu un problème majeur sur Internet. Dans le cadre de notre travail, la classification des courriers électronique est effectuée à l'aide des algorithmes d'apprentissage automatique, l'efficacité de ces classificateurs est testé avec des différentes représentations en utilisant le corpus qui nous avons créé. Nous avons crié un modèle de filtrage des spams Arabe à base de NLP, qui prédit l'origine d'un message d'entrée avec une précision de 95%. Grâce à l'utilisation de machine learning, des solutions efficaces peuvent être réalisées en vue de renforcer la capacité de détection des systèmes de détection des spams Arabe. Les résultats des tests montrent que DecisionTreeClassifier est plus performant par rapport aux algorithmes NB et KNN et SVM.

Mots-clés : Spam, Machine Learning, DecisionTreeClassifier, Sécurité Informatique, Filtre Antispam.

ABSTRACT

Email really does serve users, it's a quick and cost-effective way to exchange information. However, users quickly find themselves overwhelmed with amounts of unwanted messages also called spam. Spam quickly became a major problem on the Internet. As part of our work, the classification of e-mails is carried out using machine learning algorithms, the efficiency of these classifiers is tested with different representations using the corpus that we created. We shouted an Arabic spam filtering model based on NLP, which predicts the origin of an input message with 95% accuracy. Through the use of machine learning, effective solutions can be achieved to strengthen the detection capacity of Arabic spam detection systems. Test results show that DecisionTreeClassifier outperforms NB and KNN and SVM algorithms.

Keywords: Spam, Machine Learning, DecisionTreeClassifier, Computer Security, Spam Filter.