République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Ammar Telidji
Laghouat

Faculté des Sciences
Département de Mathématique et Informatique

Mémoire de Master

Domaine: Mathématique et Informatique
Filière: Informatique
Option: Réseaux, systèmes et applications répartis (**ReSar**)

Thème:

Etude comparative de quelques techniques de classification des SPAMs

Présenté par:
BENFERHAT KHADIDJA MAROUA

Soutenu devant le jury composé de:
Mr A. ZIYANI Université de Laghouat (Président)
Mr Y.GALLOUMA Université de Laghouat (Examinateur)
Mr Y.OUINTEN Université de Laghouat (Encadreur)

Année /……. /2011/2012

Dédicaces

Je dédie ce mémoire

A mes parents pour tous les sacrifices qu'ils m'ont accordé et leur soutien indéfini, qu'ils trouvent dans ces modestes mots ma grande estime et ma profonde reconnaissance et que dieu les bénisse.

A mes chers frères et mes sœurs qui ont été à mes cotés et qui m'ont encouragé.

A mes grands-mères, mes grands pères qui ma encouragés avec ses prières, et A tout le reste de la famille Tantes et Oncles, cousins et cousines.

A toute ma promo, tous mes amis et camarades.

Remerciements

Tout d'abord on remercie **Dieu** tout puissant pour nous avoir donné la chance, le courage, la détermination et la patience pour parvenir jusqu'ici.

En préambule à ce mémoire, je souhaite adresser ici tous mes remerciements aux personnes qui m'ont apporté leur aide et qui ont ainsi contribué à l'élaboration de ce mémoire.

Qu'il me soit permis de rendre un vibrant hommage à mon encadreur **Mr.**Y.Ouinten qui était pour moi un immense honneur de travailler avec il durant mon projet fin d'étude de master.

Je les remercie également pour leurs disponibilités, ses sens aigu de l'humanisme pédagogique, et enfin leurs rigueur intellectuelle et morale qui reste pour moi un modèle à suivre.

En fin je remercie les membres du jury qui ont bien voulu accepter, de faire partie du jury et pour procéder à l'évaluation de ce modeste travail.

Sommaire

Résumé

L’informatisation des communications a permis d’accroître la vélocité des échanges et les a considérablement enrichis en contenu. Les e-mails sont de plus en plus utilisés par les particuliers et encore plus par les entreprises. Mais comme pour le courrier traditionnel, les utilisateurs ont dû, très rapidement, faire face à des courriers non désirés, ou pourriel (SPAM en anglais), et pour la plupart tout à fait indésirables. Pour contrer ce flot de détritus et ne pas perdre les courriels qui nous sont réellement destinés, il faut automatiser la détection et la destruction de ce type de pollution numérique avec le risque qu’un document soit mal classé.

Nous présentons dans ce document une étude sur la caractérisation et la classification des spams. Nous avons effectué une comparaison de trois méthodes de classification qui sont les SVM, Naive Bayes et les arbres de décision C4.5 à l'aide du banc d'essai SpamBase de Machine Learning Repository de l'UCI (University of California Irvin) avec le logiciel de datamining TANAGRA. D'une façon générale nous avons montré que les attributs relatifs aux fréquences des mots clés ont une influence importante sur l’erreur d’apprentissage et que les classifieurs SVM et Naïve Bayes ont des performances relativement proches. Par contre C4.5 est meilleur que les deux autres.

Liste des figures

Introduction Générale

Contexte de l'étude

Le courrier électronique est aujourd'hui un des services les plus utilisés sur internet et sur les réseaux d'entreprises. Il permet une communication à la fois rapide, synchrone et économique. C'est également un moyen simple et universel d'échanges de fichiers. Ce service devenu incontournable comporte cependant de nombreux risques en termes de sécurité informatique. En effet, ces dernières années, les utilisateurs du courrier électronique du monde entier ont constaté qu'un nombre croissant de messages non sollicités (appelés spam ou pourriel) provenaient dans leurs boittes aux lettres. Les courriels indésirables, représentent une très importante part du trac mondial des courriels. Selon diverses analyses, sur les dizaines de milliard de courriels qui transitent sur le réseau quotidiennement, près de 50%[1] sont des pourriels et cette proportion ne cesse d'augmenter. Les sociétés pratiquant le publipostage électronique trouvent beaucoup d'avantages à ce mode de prospection, et notamment un avantage économique, dans le sens ou la constitution et l'envoi d'e-mails ne coute pratiquement rien. Si les couts de prospection sont faibles, les préjudices subis par les internautes sont substantiels lorsque les courriels en question n'ont pas été sollicites. C'est l'importance de ces atteintes qui touchent notamment la vie privée des internautes et se répercutent, de façon plus pragmatique, sur le prix de leur abonnement qui impose aujourd'hui que des garanties solides soient mises en place, afin de restaurer une confiance indispensable à l'épanouissement du commerce électronique, et plus largement du cyberespace.

Les deux conséquences principales du spam sont la réduction des ressources informatiques ainsi que la perte de temps pour les utilisateurs (lecture et suppression des spams). Aujourd'hui, le spam est un problème crucial pour la messagerie. A ce jour, on a proposé plusieurs méthodes de filtrage destinées à traiter ce problème telles que l'approche bayesienne, les listes noires, les listes blanches ...etc, cependant il n'existe aucune technique de filtrage capable d'offrir une protection complète et sûre.

[1]Source : Radicati Group, 2003. Estimations pour 2007.

L'objectif de notre travail

L'objectif de notre travail est dans un premier temps d'étudier la caractérisation et la classification des spams et dans un deuxième temps choisir deux où trois méthodes de classification de spams et de comparer leurs performance en utilisant un banc d'essai connu. A défaut d'implémenter les méthodes choisies nous aurons à choisir un logiciel qui propose une implémentation de ces dernières.

L'organisation du mémoire

Le présent mémoire est organisé en quatre chapitres et une conclusion. Après une brève introduction générale du sujet, le premier chapitre introduit le phénomène du courriel indésirable également appelé spam. Nous y donnons une définition du spam, ainsi que les différents types de spam. Nous y exposons aussi les effets du spam ainsi que les moyens de lutte et de protection contre le spam.

Le deuxième chapitre est consacré à la catégorisation de textes et ses domaines d'applications. Nous y exposons les différents algorithmes d'apprentissage utilisés pour la catégorisation de textes. Le troisième chapitre contient une présentation de l'application des classifieurs à la détection de spam ainsi que les outils d'analyse open source. Le quatrième chapitre est consacré à l'étude expérimentale. Nous y exposons une description détaillée de la base de données utilisée, et les résultats obtenus de la comparaison des performances des trois méthodes choisies ainsi qu'une discussion sur ces derniers. Le mémoire se termine par une conclusion générale.

Chapitre 1

Le Spam : Le courriel indésirable

Introduction

Depuis quelques années, nos boites aux lettres électroniques se retrouvent quotidiennement remplies de courriels non sollicités. Chaque jour, d'énormes quantités d'emails, à caractère publicitaire ou frauduleux, inondent nombre d'entreprises et de particulier. Notre messagerie se retrouve aujourd'hui attaquée par des commerciaux peu scrupuleux. Ce sont ces-emails, le plus souvent à connotation vaguement commerciale, publicitaire, que l'on appelle des «spams».

Le spam peut s'attaquer à divers medias électroniques : les courriels, les forums de discussions, les moteurs de recherche, les wikis, les messageries instantanées. Le courrier électronique indésirable appelé pourriel ou spam est le plus répandu. Le cout d'envoi d'un courrier électronique étant négligeable, il est facile d'envoyer un message à des millions de destinataires. Les destinataires assument le cout de réception et de stockage en boites aux lettres, ce qui peut causer des couts non négligeables aux prestataires de services, à cause du volume pris par le pourriel qui lui est considérable : 93 % des courriers reçus au printemps 2007 étaient des spams[2]. Contrairement aux promotions commerciales pour lesquelles les utilisateurs peuvent avoir donné leur accord, le spam n'est pas sollicite.

Le premier envoi de masse de messages publicitaires date du 3 mai 1978. Il fut envoyé par un certain Gary Thuerk, spécialiste marketing au sein de la société DEC. Il envoya son message auprès de la totalité des utilisateurs du réseau vivant sur la côte ouest des Etats-Unis, soit environ 600 personnes[3]. Malheureusement, sa maitrise insuffisante du système d'envoi de mails a fait qu'une bonne partie des adresses des destinataires apparaissait en fait dans le corps du message et, qu'en conséquence, ceux-ci ne peuvaient recevoir le courriel. En constatant cela, Gary Thuerk réexpédie aussitôt le message à plusieurs reprises sans pour autant prendre conscience de sa bévue. Ce premier spam, historiquement parlant, lui couta son poste[4].

Le spam est aujourd'hui omniprésent sur toutes les messageries de la planète et envahit littéralement nos serveurs puis nos disques durs. Cela provoque des ralentissements du trafic Internet et accroit les risques de virus informatiques.

C'est la raison pour laquelle on assiste, depuis quelques années, à la mise en place d'une riposte nécessaire : la lutte anti-spam. Les techniques pour lutter contre le spam mettent en œuvre diverses techniques de classification automatique pour distinguer le spam du courrier

[2]Source : Radicati Group, 2003. Estimations pour 2007.

[3] http : //linuxfr.org/2003/06/23/12931.html

[4]http://www.arobase.org/culture/premierspam.htm

légitime. Ces moyens, principalement de nature technique, sont devenus un enjeu commercial considérable, à tel point qu'ils font aujourd'hui partie intégrante des politiques de sécurité des réseaux. Ces outils anti-spam combinent couramment plusieurs techniques de lutte, allant des simples listes d'adresses d'expéditeurs connus à des techniques plus complexes issues de statistiques et d'heuristiques. Dans le reste de ce chapitre, nous allons donner quelques définitions ainsi que quelques types de courriels indésirables.
Nous exposerons aussi les effets du spam ainsi que les moyens de protection disponibles.

1.1 Définition

Il n'existe pas de définition officielle et universelle du mot "spam". Néanmoins, la CNIL (Commission Nationale Informatique et Libertés) le décrit ainsi : "Envoi massif et parfois répété de courriers électroniques non sollicités à des personnes avec lesquelles l'expéditeur n'a jamais eu de contact au préalable, et dont il a capté l'adresse électronique de façon irrégulière " [15].
Aussi appelé pourriel, polluriel, courrier-rebut, il s'agit à l'origine d'une marque anglaise de corned-beef. Plus précisément, SPAM est un acronyme pour "Spiced Pork And Meat" (paté épicé à base de porc et de viande). La théorie la plus courante veut que le terme provienne d'un sketch des Monty Python, dans lequel les comiques britaniques chantaient: "Spam spam spam spam , spam spam spam spam , spam spam spam...". La chanson, interminable et interprétée crescendo, couvrait les propos des autres protagonistes [7].
Le spam est parfois décrit de façon moins restrictive. C'est par exemple le cas de la définition suivante publiée en 1997 par Eric Demeester sur son site personnel : " Message dont le mode de diffusion et/ou le contenu sont nuisibles pour les réseaux et/ou pour les lecteurs " [7].
Pour résumer, l'expéditeur d'un message de spam poursuit l'une des tâches suivantes: Faire de la publicité des biens, des services ou des idées ; tromper les utilisateurs pour qu'ils fournissent leur information privées ; fournir des logiciels malveillants ; ou bloquer temporairement un serveur de messagerie. Les contenus des spams couvrent divers sujets et sont de types très différents. Ceci fait qu'ils peuvent simuler différents types de courrier légitime, comme des memos, des lettres, et confirmations de commande [15].

1.2 Les différents types de courriels indésirables

Déterminer l'évolution précise des contenus véhiculés au cours du temps n'est pas chose aisée pour plusieurs raisons :

- Les contenus varient fortement plusieurs fois par année, parfois de façon contradictoire. Effectivement, il n'est pas rare de constater qu'un type de message atteint un pic pendant quelques mois, rattrapé quelques temps plus tard par un autre, au gré des spammeurs qui changent de registre au fur et à mesure des opportunités qui s'offrent à eux suivant l'actualité (nouveaux médicaments, situation économique, etc...) ou des tendances actuelles de consommation.
- Les contenus sont très différents en fonction de la langue, mais aussi de la cible (particuliers ou entreprises).
- Les honeypots utilisés par les différentes sociétés ou organismes ne sont pas tous spammés de façon identique (rien ne leur garantit qu'ils détiennent un échantillon hétérogène des types de spams émis partout dans le monde).
- Les messages sont catégorisés de façon automatique, ce qui peut conduire à des inexactitudes plus ou moins marquées.

Le spam contient généralement de la publicité ou des propositions malhonnêtes provenant d'escrocs de tout genre. Les lettres en chaine peuvent être qualifiées de spam. Mais, ce n'est plus le seul objectif des spammeurs. Ils ont désormais recours à d'autres méthodes bien plus répréhensibles. Ces orientations ne sont pas nouvelles, mais nous constatons qu'elles sont toujours plus marquées mais aussi plus subtiles et mieux rodées qu'auparavant. Citons par exemple l'explosion du spam boursier entre 2005 et 2006, l'augmentation des revenus issus du phishing en 2007[5].

Dans la **Figure 1.1** ci-dessous, nous présentons un panorama des principaux types de contenus véhiculés :

Les courriels indésirables peuvent revêtir divers aspects ; la description de ces courriels est présentée dans la **Figure 1.2** ; voici quelques déclinaisons possibles :

[5]Source : Radicati Group, 2003. Estimations pour 2007.

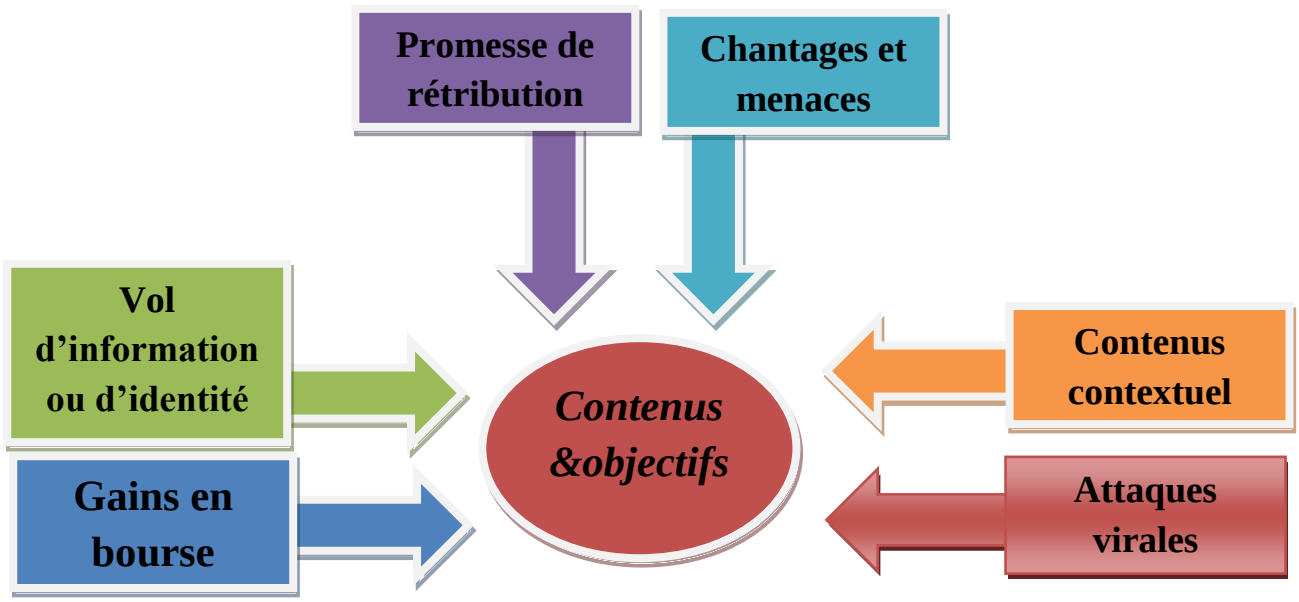


Figure 1.1 - Un panorama des principaux types de courriels indésirables [15].

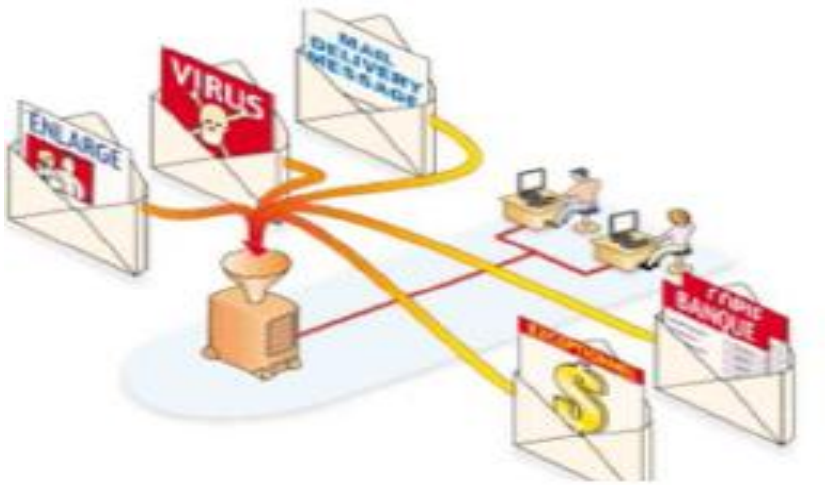


Figure 1.2 - Les courriels indésirables existants [7].

- ***Spams commerciaux (UCE/UBE) :***

Il s'agit d'e-mails publicitaires vantant l'intérêt d'un produit ou d'un service. Généralement rédigés en anglais, ils représentent la majeure partie des courriers indésirables envoyés. Ces spams concernent essentiellement la santé, les loisirs, la finance et les services web, mais peuvent également présenter un contenu choquant (messages à caractère pornographique, politique, religieux, racial...). On distingue les UCE (Unsolicited Commercial E-mail) des UBE (Unsolicited Bulk E-mail). Les premiers sont envoyés par des entreprises ne respectant pas, intentionnellement ou par méconnaissance des législations, les réglementations pour leurs envois (envois massifs, collectes suspectes, liens de désinscription absents ou inactifs...). Les UBE sont beaucoup plus pernicieux. Ils sont envoyés de sombres façons par des spammeurs professionnels. Ceux-ci inventent de nombreuses techniques pour déjouer les règles des solutions anti-spam et savent généralement bien conserver l'anonymat.

- ***Virus à propagation automatique :***

Ils exploitent les carnets de contacts des machines infectées et se propagent avec une facilité extrême. Ils sont souvent inclus dans un e-mail simple contenant quelques mots et une pièce jointe. De surcroît, les créateurs de virus travaillent de plus en plus main dans la main avec les spammeurs, contribuant ainsi à leur diffusion.

- ***Notifications de serveurs :***

Egalement appelés "Mail Delivery Message", ces e-mails sont envoyés automatiquement sur l'adresse de l'expéditeur pour le prévenir qu'un destinataire n'a pas reçu son message.
Avec la prolifération des virus utilisant les carnets de contacts, ces messages sont de plus en plus nombreux.

- ***Scam :***

C'est une escroquerie qui repose le plus souvent sur des propositions de participation à une opération financière internationale très alléchante. Initialement pratiqués par courrier traditionnel ou par fax, les scams ont aujourd'hui leur déclinaison par e-mail.

- ***Phishing :***

Cette technique consiste à prendre l'apparence visuelle d'un service en ligne connu et à demander à un internaute réellement client du site de mettre à jour ses données personnelles dans un formulaire factice afin de les intercepter. Le montant total des fraudes attribuables au phishing en 2004 s'élève à 137 millions de dollars, occasionnés par 31 000 attaques [2].

1.3 Combien coûte le spam

Le spam coûte beaucoup d'argent aux entreprises. Il est estimé entre 600 et 1000 dollars par an et par salarié [2]. En fonction du nombre de postes et de la quantité moyenne de spams reçus, il est assez simple de réaliser une estimation du coût généré par le spamming pour une entreprise. Cette charge inutile nuit au bon fonctionnement de l'entreprise : elle paralyse l'activité des employés et monopolise les ressources informatiques utiles à d'autres tâches. Le spam en quelques chiffres [2] :

- 100 : croissance du coût du spam chaque année.
- 42 milliards de $: coût global pour les entreprises au niveau mondial en 2004 (prévision : 200 milliards de $ en 2007).
- 600 à 1000 $: coût par an et par salarié.
- Plus des 2/3 du volume total et mondial d'e-mails envoyés.
- 85% des spams reçus en France sont rédigés en langue anglaise (7% en français)
- 60% proviennent des Etats-Unis

1.4 Les effets du spam

Les inconvénients majeurs du spam sont :

- l'espace qu'il occupe dans les boites aux lettres des victimes.
- la difficile consultation des messages personnels ou professionnels au sein de nombreux messages publicitaires et l'augmentation du risque de suppression erronée ou de non-lecture de messages importants.
- la perte de temps occasionnée par le tri et la suppression des messages non sollicités.
- le caractère violent ou dégradant des textes ou images véhiculés par ces messages, pouvant heurter la sensibilité des plus jeunes ;
- la bande passante qu'il gaspille sur le réseau des réseaux (Internet, réseau WAN).

Le spam induit également des coûts de gestion supplémentaires pour les fournisseurs d'accès à internet (FAI), se répercutant sur le coût de leurs abonnements. Ce surcoût est notamment lié à [7] :

- ✓ la mise en place des systèmes anti-spam.
- ✓ la sensibilisation des utilisateurs.
- ✓ la formation du personnel.
- ✓ la consommation de ressources supplémentaires (serveurs de filtrage, etc...).

1.5 Les moyens de protection

1.5.1 Eviter le spam

Afin d'éviter le spam, il est nécessaire de divulguer son adresse électronique le moins possible et à ce titre :

- Ne pas relayer les messages (blagues, etc...) invitant l'utilisateur à transmettre le courrier au maximum de contacts possible. De telles listes sont effectivement des aubaines pour les collecteurs d'adresses. Il est éventuellement possible de faire suivre le message en s'assurant de masquer les adresses des destinataires précédents.
- Eviter au maximum de publier son adresse électronique sur des forums ou des sites internet.
- Dans la mesure du possible remplacer son adresse électronique par une image (non détectable par les aspirateurs d'adresses) ou bien en la décomposant.
- Créer une ou plusieurs " adresses-jetables " servant uniquement à s'inscrire ou s'identifier sur les sites jugés non dignes de confiance.

1.5.2 La nécessité du filtrage du spam

L'application d'un filtre anti-spam fiable devient de plus en plus importante pour les internautes car ils ont à faire face à la quantité croissante d'e-mails non sollicités. Différentes méthodes très simples ont été découvertes au cours de l'évolution du filtrage des spams :

1.5.2.1 Architecture réseau - Vérificateur et proxy :

Indépendamment de la technologie anti-spam centrale, la conception d'une solution anti-spam peut se faire de deux manières. Certains produits anti-spam sont conçus comme étant des «vérificateurs» tandis que d'autres sont conçus comme des «proxys». Avec les vérificateurs, il faut d'abord utiliser le produit anti-spam pour vérifier la boite de réception et supprimer les spam. Les «proxys» se logent entre le serveur de mail et la boite de réception, et filtre les spam de façon transparente.

Le scénario «proxys», appelé «filtrage anti-spam à la demande», apporte des avantages significatifs par rapport au scénario «vérificateur».

1.5.2.2 Listes noires (Blacklist) :

Il y a un besoin, à faire la différence entre les niveaux de listes noires :

Listes noires personnalisées

La liste noire personnalisée est une base de données d'adresses emails et de domaines dont on ne désire pas recevoir de messages. Les messages provenant de ces adresses ou domaines seront toujours marqués comme spam. L'inconvénient de cette liste est qu'il faut l'alimenter manuellement à la détection d'une nouvelle adresse mail émettrice de spam, ou bien la mettre à jour régulièrement auprès d'une base de données centralisée afin de toujours posséder la dernière version de la liste des expéditeurs bloquées.

Listes noires DNS(DNSBL)

Les listes noires DNS sont des bases de données de serveurs SMTP qui ont été utilisées pour lutter contre le spam. Quand un email est envoyé, il passe par un certain nombre de serveurs SMTP jusqu'à ce qu'il atteigne la destination finale. L'adresse IP de chacun de ces serveurs SMTP est enregistrée dans l'entête de l'email. Il suffit donc pour chaque email reçu de vérifier l'adresse IP trouvée dans l'entête du message avec une base de données DNSBL. Les administrateurs de messagerie doivent donc configurer leurs serveurs de façon à interroger cette source d'informations et prendre une décision quant aux messages provenant des sources listées.

Listes maintenues en temps réel(RBL)

Les RBL sont un type de listes noires organisées en temps réel par certains fournisseurs d'accès. Ces listes gèrent l'inscription et retraits des machines identifiées comme des «open relays». La gestion automatique est rendue nécessaire par le caractère «temps réel» de ces listes qui sont mises à jour de seconde en seconde en fonction des plaintes et des tests effectués. Il arrive malheureusement que des serveurs soient blacklistés à tort. En fonction de la RBL utilisée, le risque de générer des faux positifs est plus ou moins élevé.

1.5.2.3 Listes blanches

Par opposition aux listes noires, les listes blanches contiennent les adresses email des expéditeurs de confiance. Dans cette méthode, le risque de faux positifs est totalement écarté mais l'inconvénient est que le destinataire doit traiter manuellement les messages provenant des expéditeurs inconnus.

1.5.2.4 Analyse heuristique

L'analyse heuristique est composée de critères permettant d'examiner l'entête et le corps d'un message afin d'extraire des caractéristiques laissant penser qu'il s'agit ou non d'un spam[6].

[6] http ://www.secuser.com/dossiers/methodes antispam.htm

Il résulte de cette analyse l'établissement d'un score, s'il dépasse un certain seuil, le message est considéré comme un spam. Le moteur d'analyse heuristique est habituellement composé de plusieurs centaines de critères représentés sous forme d'expressions régulières. Le choix du seuil à partir duquel le message est considéré comme spam est influant sur les résultats atteints.

1.5.2.5 Filtre sur empreinte (bases collaboratives de spam)

Le filtre sur empreinte fonctionne selon le même principe que les anti-virus. Comme son nom l'indique, une empreinte est réalisée sur le spam par une fonction de hachage, puis elle est stockée dans une base de données. Le terme «base collaborative» désigne l'exploitation d'une même base de données par différents usagers. L'un des avantages de cette technique est qu'elle nécessite peu de ressources. Des messages différents peuvent avoir la même empreinte, mais la probabilité que cela arrive est extrêmement faible, donc le risque de produire des faux positifs est très faible dans la pratique.

1.5.2.6 Nouvelles techniques

Il s'agit généralement de systèmes possédant des filtres sur la base d'un apprentissage. Cette nouvelle méthode donne la possibilité pour les ordinateurs de prendre leurs propres décisions. L'apparition de techniques d'apprentissage dans le filtrage de spam a effectué une amélioration significative de filtrage. Les spammeurs ne sont plus en mesure de tester les filtres avant d'envoyer les messages, car le filtre de chaque utilisateur a sa propre base de connaissances. Plusieurs modèles ont été mis en place ces dernières années, on propose d'introduire les plus répandus :

Filtre bayésien

Le filtre bayésien se base sur des calculs statistiques liés aux mots clés habituellement rencontrés dans les spams ou les messages légitimes.

Les prédictions sont réalisées sur la base des expériences passées[7].

Le filtre bayésien calcule la probabilité que le nouveau message soit retenu comme spam ou non.

L'avantage majeur de ce filtre est qu'il s'adapte de lui-même aux besoins. Cette approche bayésienne est très efficace et difficile à contourner, un article de mai 2003 de la BBC a signalé que des taux de détection de spam de plus de 99,7% peuvent être réalisés avec un nombre très bas de faux positifs[8].

[7] http ://www.lesnouvelles.net/articles/produits/spamchallenge-response-pire-que-le-mal

[8] http ://www.lesnouvelles.net/articles/produits/spamchallenge-response-pire-que-le-mal

SpamAssassin

SpamAssassin est une solution anti-spam open-source développée par l'Apache Software Foundation (éditrice du serveur web Apache). Fortement adaptable, SpamAssassin peut être installé sur divers serveurs de messagerie, tant sur Linux (cas le plus courant) que sur Windows, voir même sur Mac OS X[9].

Cependant, cette solution est nettement plus difficile à installer que la plupart des autres anti-spams et la qualité du filtrage n'est habituellement pas optimale avec la configuration par défaut. SpamAssassin est réputé pour consommer beaucoup de ressources. Sa mise en œuvre sera probablement plus délicate dans les entreprises traitant de grands volumes de messages.

1.6 Conclusion

Ces technologies alternatives ne se présentent pas comme des solutions infaillibles dans la lutte contre le spam. Bien qu'elles soient pour l'instant adoptées par une minorité de sites, leur développement et leur adoption à une plus grande échelle devraient à terme les amener à compléter la boite à outils anti-spam, afin de la rendre encore plus étanche à ce phénomène.

Cependant, il serait souhaitable qu'un standard unique puisse émerger afin d'éviter la collision de trop nombreuses technologies au détriment du but recherché et des performances des serveurs de messagerie et des réseaux. Cependant, les utilisateurs d'Internet se retrouvent assez vite submergés de quantités astronomiques de courriels indésirables dont le traitement nécessite un temps considérable. Devant l'importance de ce phénomène, il est donc nécessaire d'élaborer des outils efficaces capables de traiter et de filtrer le courriel ce qui est présenté dans le chapitre suivant.

[9]http ://fr.wikipedia.org/wiki/Spam-assassin

Chapitre 2

Catégorisation des documents

Introduction

Nous avons vu dans le chapitre précédent, que le courrier électronique est le mode de communication le plus populaire. Il est devenu un moyen rapide et économique pour échanger des informations. Cependant, les utilisateurs d'Internet se retrouvent assez vite submergés de quantités astronomiques de courriels indésirables dont le traitement nécessite un temps considérable. Devant l'importance de ce phénomène, il est donc nécessaire d'élaborer des outils efficaces capables de traiter et de filtrer le courriel.

L'apparition de techniques de classification automatique dans le filtrage de spam a effectué une amélioration significative de filtrage.

Dans ce chapitre nous abordons la notion de catégorisation de documents qu'on appelle aussi classification supervisé des documents.

C'est sur cette représentation qu'on peut utiliser les techniques de catégorisation pour déterminer la classe d'un document. Des critères d'évaluation de la catégorisation sont cités à la fin du chapitre.

2.1 Objectif de la catégorisation des documents

La catégorisation automatique des documents est une des tâches classiques de la recherche d'information et, en tant que telle, elle a suscité de nombreuses études depuis relativement longtemps. On retrouve des travaux portant sur ce sujet depuis au moins le début des années 1960. Mais la recherche dans ce domaine est toujours très pertinente, car les résultats obtenus aujourd'hui sont encore sujets à amélioration. La catégorisation de documents est l'activité du Traitement Automatique des Langues.

Cette activité est essentielle dans de nombreux domaines économiques : elle permet d'organiser des corpus documentaires, de les trier, et d'aider à les exploiter dans des secteurs tels que l'administration, l'aéronautique, la recherche sur internet, les sciences. La catégorisation de textes peut être un support pour différentes applications parmi lesquelles le filtrage, qui consiste à déterminer si un document est pertinent ou non (décision binaire), par exemple la détection de spams (les courriers indésirables) pour ensuite les supprimer, le routage qui permet d'affecter un document à une ou plusieurs catégories parmi *n* [9].

Chacun des types de textes possède des particularités qui rendent la tâche de catégorisation plus ou moins ardue [20]. Un système de catégorisation repose sur plusieurs étapes. On peut les schématiser :

1. l'extraction d'attributs pertinents, consiste à rechercher un nouvel ensemble d'attributs (aussi appelés variables, traits ou descripteurs) dérivé de la description initiale des données, et conservant un maximum d'information sur ces données ;

2. la phase d'apprentissage à partir d'un corpus d'entrainement est généralement assurée par des algorithmes de classification et

3. l'évaluation du classificateur sur un corpus test, nécessite le recours aux mesures de performance telles que la précision et le rappel.

Ces trois étapes constituent chacune un sous-domaine de recherche à part entière [1].

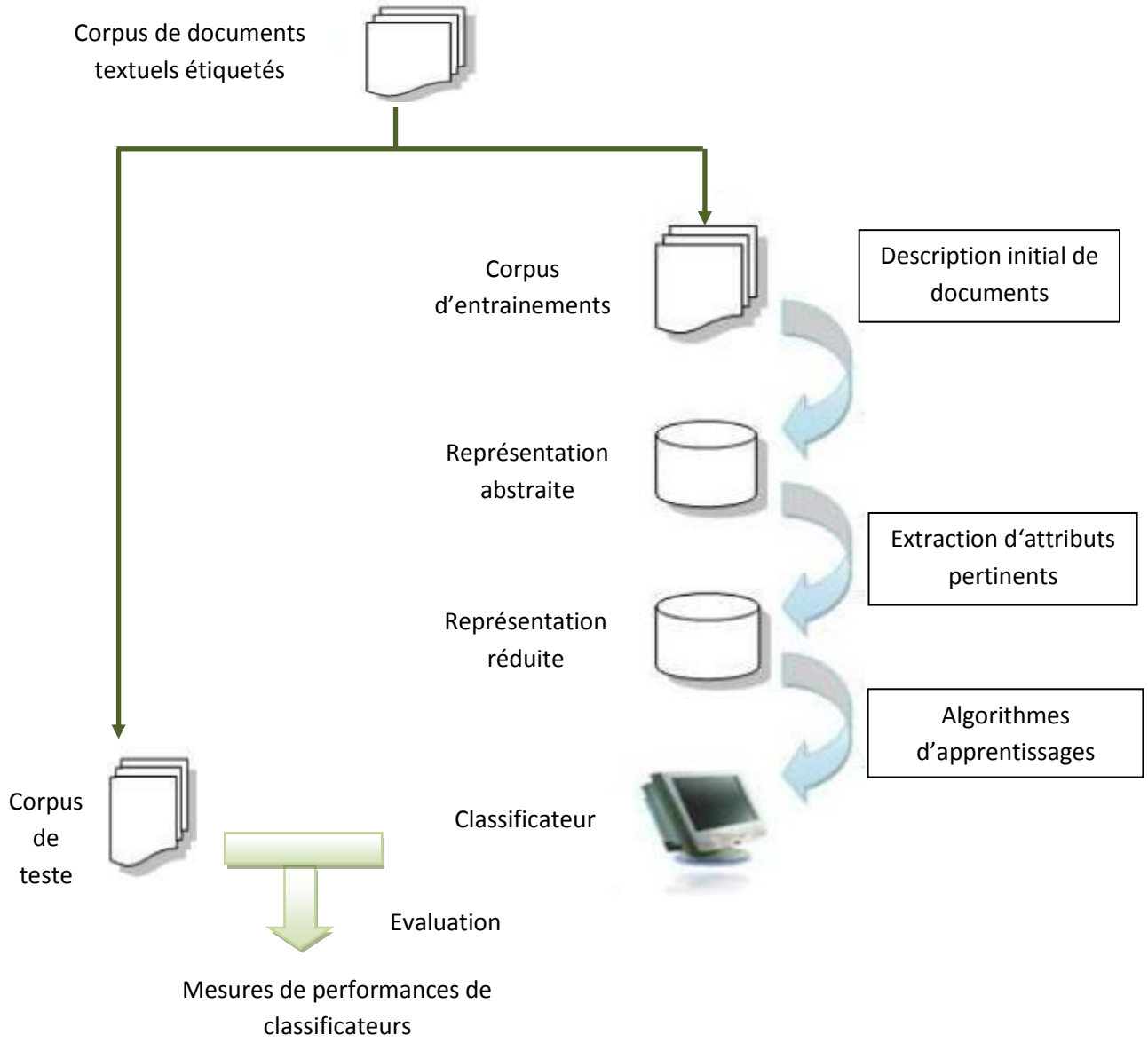


Figure 2.1 - Processus de catégorisation automatique de textes [1].

2.2 Représentation des données textuelles

Les algorithmes d'apprentissage ne sont pas capables de traiter directement les textes, plus généralement, les données non structurées comme les textes, les images, les sons et les séquences vidéo. C'est pourquoi une étape préliminaire dite représentation est nécessaire. Cette étape consiste généralement en la représentation de chaque document par un vecteur, dont les composantes sont par exemple les mots contenus dans le texte, afin de le rendre exploitable par les algorithmes d'apprentissage. Une collection de textes peut être ainsi représentée par une matrice dont les lignes sont les termes qui apparaissent au moins une fois et les colonnes sont les documents de cette collection [9].

Ainsi, le choix d'un mode de représentation adéquat des instances à traiter est une étape incontournable en apprentissage automatique : on doit opter pour une façon uniforme et judicieuse d'abstraire les données avant de les soumettre à un algorithme. Comme l'apprentissage joue un rôle dans la catégorisation automatique de textes, le choix d'un mode de représentation y devient un enjeu. Par la suite, il sera question de la sélection d'attributs, presque toujours impliqués en catégorisation automatique de textes et on éliminera les attributs jugés inutiles à la classification [20].

Un grand nombre de chercheurs dans le domaine ont choisi d'utiliser une représentation vectorielle dans laquelle chaque texte est représenté par un vecteur de *n* termes pondérés. A la base, les n termes sont tout simplement les n différents mots apparaissant dans les textes de l'ensemble d'entraînements.

2.2.1 Représentation en «sac de mots»

La représentation de textes la plus simple a été introduite dans le cadre du modèle vectoriel elle porte le nom de «sac de mots». L'idée est de transformer les textes en vecteurs dont chaque composante représente un mot. Cette représentation des textes exclut toute analyse grammaticale et toute notion de distance entre les mots [9].

2.2.2 Représentation des textes par des phrases

Malgré la simplicité de l'utilisation de mots comme unité de représentation, certains auteurs proposent plutôt d'utiliser les phrases comme unité. Les phrases sont plus informatives que les mots seuls, car les phrases ont l'avantage de conserver l'information relative à la position du mot dans la phrase.

Logiquement, une telle représentation doit obtenir de meilleurs résultats que ceux obtenus via les mots. Cependant, les expériences menées ont montré que les résultats n'étaient pas meilleurs [4] et que surtout ces approches sont très difficile à mettre en œuvre. Les qualités sémantiques sont conservées. Les qualités statistiques sont largement dégradées [9].

2.2.3 Représentation des textes avec des racines lexicales et des lemmes

Dans le modèle précédent (représentation en «sac de mots»), chaque flexion d'un mot est considérée comme un descripteur différent et donc une dimension de plus ; ainsi, les différentes formes d'un verbe constituent autant de mots. Par exemple : les mots «déménageur, déménageurs, déménagement, déménagements, déménager, déménage, déménagera, etc.» sont considérés comme des descripteurs différents alors qu'il s'agit de la même racine «déménage» ; les techniques de désuffixation (ou stemming), qui consistent à rechercher les racines lexicales, et de lemmatisation cherchent à résoudre cette difficulté. Pour la recherche des racines lexicales, plusieurs algorithmes ont été proposés ; les algorithmes de «stemming» les plus connus sont ceux de Lovins [12] et de Porter [16].

La lemmatisation permet le regroupement des formes morphologiques d'une même unité linguistique en une seule unité appelé lemme. Elle réduit ainsi des mots en entités premières, appelées lemmes ou formes canoniques [14].

2.2.4 Pondération des termes

Une fois que l'on choisit les composantes du vecteur représentant un texte *j*, il faut décider de la façon d'associer un poids à chaque coordonnée de son vecteur *dj*. Il existe différentes méthodes pour calculer le poids w_{KJ}. Ces méthodes sont basées sur les deux observations suivantes [22] :

- Plus le terme t_k est fréquent dans un document d_j plus il est en rapport avec le sujet de ce document.
- Plus le terme t_k est fréquent dans une collection, moins il sera utilisé comme discriminant entre documents.

2.3 Réduction des dimensions

Les modèles de classification de texte, considèrent, pour la plupart, des documents avec des représentations «sac de mots» qui ne prennent pas en compte l'ordre des mots dans les documents ni leur structure, de telle sorte que le vocabulaire apparaissant dans le corpus

d'entrainement constitue l'ensemble initial d'attributs. En effet, plus l'espace de représentation est de dimension élevée, plus il y a de paramètres à estimer, et la taille de la base d'apprentissage doit évoluer en conséquence. Chaque composante du vecteur d'attributs correspondant à un terme, des espaces à plusieurs milliers de dimensions sont rapidement atteints. La taille de l'espace de description est alors très importante et donc limitative (en terme de complexité) pour l'utilisation de techniques de classification classiques. De plus, beaucoup d'attributs composant ce vocabulaire sont peu discriminants et peu pertinents, ajoutons à cela le fait que la matrice (documents × *mots)* est très éparse (beaucoup de 0). Par conséquent, le problème d'extraction d'attributs pertinents pour la classification de documents est fortement lié à la nécessité de réduire la dimension et d'améliorer la pertinence de l'espace de description des documents. Autres raisons qui nécessitent la réduction de l'espace : un modèle plus simple sera plus facile à comprendre et à interpréter ; le déploiement sera facilité, nous aurons besoin de moins d'informations à recueillir pour la prédiction ; enfin, un modèle simple se révèle souvent plus robuste en généralisation c.-à-d. lorsqu'il est appliqué sur la population. Pour cela, plusieurs techniques ont été mises en place pour réduire la dimension du vocabulaire, des techniques qui se divisent en deux grandes familles [25] :

2.3.1 Sélection d'attributs

La sélection d'attributs prend les attributs (ou mots) d'origine et conserve seulement ceux jugés utiles à la classification, selon une certaine fonction d'évaluation, en attribuant un score à chaque terme en fonction de son pouvoir discriminant : un terme fortement discriminant du point de vue de la classification thématique aura un score plus élevé qu'un terme moins discriminant. Il suffit alors de ne conserver que les termes dont le score est le plus élevé pour réduire considérablement la dimension de l'espace. Présente une étude comparative de cinq mesures de sélection d'attributs utilisées dans les systèmes de catégorisation. Nous présentons les trois mesures les plus utilisées lors de la sélection d'attributs [25].

1. Le gain d'information «*IG*» : est directement issu de la théorie de l'information. Il mesure la quantité d'information apportée par la connaissance de l'apparition ou non d'un terme dans le processus de décision. Soit c_1, c_2,..., c_k l'ensemble des catégories. Le gain d'information *IG*(*t*) apporté par un terme *t* est défini comme suit :

$$IG(t) = -\sum_{i=1}^{k} P(c_j) log P(c_j) + P(t) \sum_{i=1}^{k} P(c_j|t) log(P(c_j|t) + P(\vec{t}) \sum_{i=1}^{k} P(c_j|\vec{t}) log\, P(c_j|\vec{t})$$

$P(c_j)$ peut être estimée par la proportion de documents de la base appartenant à la classe c_j et P(t) comme étant la proportion de documents contenant le terme t. De même, $P(c_j|t)$ peut être calculée par la fraction de documents de la classe c_j contenant le terme *t*.

Enfin, $P(c_j|t)$ est la fraction de documents de la classe c_j ne contenant pas le terme *t*. Il suffit alors de ne conserver que les termes ayant le meilleur score.

2. La statistique du χ^2 : mesure la corrélation entre un terme *t* et une catégorie *c*. Elle est définie comme suit :

$$\chi^2(t,c) = N \times \frac{[P(t,c)P(\vec{t},\vec{c}) - P(\bar{t},c)P(t,\bar{c})]^2}{P(c)P(\bar{c})P(t)P(\bar{t})}$$

3. L'information mutuelle «*MI* » : mesure la quantité d'information apportée par la connaissance de l'apparition d'un terme dans une certaine catégorie. La faiblesse de cette mesure est qu'elle est beaucoup trop influencée par la fréquence des mots. Pour une même probabilité conditionnelle sachant la catégorie, un terme rare va être avantagé, car il risque moins d'apparaître en dehors de la catégorie.

$$\text{MI (t,c)} = \log \frac{P(t|c)}{P(t)P(c)}$$

2.4.2 Extraction d'attributs

A partir des termes de départ, les mesures créent de nouveaux termes, en faisant soit des regroupements ou des transformations. Plusieurs techniques ont été mises en place la plus connue est sous le nom de Latent Semantic Indexing (LSI) [23] [21], consiste, en revanche, à définir un nouvel ensemble de termes, chaque nouveau terme étant construit par combinaison linéaire des termes initiaux. Le regroupement de termes («term clustering») fait aussi partie de ces techniques et a pour but de grouper les termes qui ont une sémantique commune. Les groupes « clusters » ainsi créés deviennent les termes d'un nouvel espace vectoriel.

2.4 Apprentissage automatique en catégorisation de documents

C'est principalement par apprentissage automatique que l'on tente de résoudre le problème de la catégorisation automatique de textes. Dans cette optique, plusieurs algorithmes mis au point pour des problèmes quelconques en apprentissage automatique ont été adaptés et appliqués dans ce domaine de recherche. Mais, on doit d'abord définir la notion d'apprentissage automatique.

L'apprentissage automatique fait référence au développement, à l'analyse et à l'implémentation de méthodes qui permettent à une machine (au sens large) d'évoluer grâce à un processus

d'apprentissage, et ainsi de remplir des tâches qu'il est difficile ou impossible de remplir par des moyens algorithmiques plus classiques.

L'apprentissage est aussi un des domaines clés de l'Intelligence Artificielle. L'étymologie accorde à l'être intelligent la faculté d'association. Être intelligent, au sens premier du terme, c'est pouvoir repérer ou créer des liens entre des objets. C'est une vision plutôt synthétique de l'Intelligence. On trouve aujourd'hui dans ce mot par extension bien d'autres connotations : rapidité, adaptabilité, faculté d'analyse, aptitude à apprendre ... etc.

En fait parler d'intelligence pour des êtres humains ou des machines ne peut se faire que par référence à l'homme. Turing l'a bien compris, lorsqu'il a défini son fameux test : une machine est intelligente si son comportement ressemble à celui de l'homme à s'y tromper. En effet, l'Intelligence Artificielle a pour but de reconstituer, à l'aide de moyens artificiels, des raisonnements et des actions intelligentes et ce par apprentissage [11].

Autrement dit, l'apprentissage automatique est le champ d'étude où l'on essaie de mimer et de reproduire la capacité de l'homme à apprendre. L'apprentissage peut être de deux types :

2.4.1 Apprentissage non supervisé

Il n'y a pas de classes prédéfinies et le but est d'effectuer les meilleurs regroupements possibles entre les objets dans lesquels les observations diffèrent très peu au regard de ses valeurs.

L'apprentissage non supervisé, appelé aussi apprentissage par observations ou par découverte, consiste à trouver des «régularités» dans l'échantillon d'apprentissage.

2.4.2 Apprentissage supervisé

L'apprentissage supervisé est une technique d'apprentissage automatique où l'on cherche à produire automatiquement des règles à partir d'une base de données d'apprentissage contenant des exemples de cas déjà traités.

Plus précisément, la base de données d'apprentissage est un ensemble de couples entrée-sortie $(x_n;y_n)$; $1 \leq n \leq N$ avec $x_n \in X$ et $y_n \in Y$, que l'on considère être tirées selon une loi inconnue.

La méthode d'apprentissage supervisé utilise cette base d'apprentissage afin de déterminer une représentation compacte de g et appelée fonction de prédiction, qui à une nouvelle entrée x associe une sortie g (x). Le but d'un algorithme d'apprentissage supervisé est donc de généraliser pour des entrées inconnues ce qu'il a pu «apprendre» grâce aux données déjà traitées par des experts. On distingue généralement deux types de problèmes que l'on cherche à résoudre avec une méthode d'apprentissage automatique supervisée :

- $Y \subseteq \mathbb{R}$: Lorsque la sortie que l'on cherche à associer à une entrée est une valeur dans un ensemble continu de réels, on parle d'un problème de régression.
- $Y = 1,\ldots,I$: Lorsque l'ensemble des valeurs de sortie est de cardinal fini, on parle d'un problème de classification car le but est en fait d'attribuer une étiquette à une entrée donnée.

2.5 Les algorithmes d'apprentissage supervisé

Dans le courant de l'apprentissage automatique, différents types de classifieurs ont été mis au point, toujours dans le but d'atteindre un degré maximal de précision et d'efficacité, chacun ayant ses avantages et inconvénients.

Parmi les classifieurs existants, on peut faire des regroupements et distinguer des grandes familles. Par exemple, on peut discerner les classifieurs probabilistes qui utilisent l'ensemble d'entraînement, c'est-à-dire les textes déjà classés, pour estimer les paramètres de la distribution de probabilité des mots par rapport aux catégories. C'est dans cette famille qu'on retrouve entre autres le classifieur bayésien naïf. On trouve aussi des classifieurs se basant sur un profil, les classifieurs linéaires. Dans ce contexte, le profil est un vecteur de termes pondérés construit pour chaque catégorie, dans le but de les représenter d'une façon générale. Ce vecteur est bien sûr construit à l'aide des données d'entraînement. Quand un nouveau texte doit être classé, il est alors comparé à ce vecteur «type». Un avantage de cette approche est qu'elle produit un classifieurs compréhensible par un humain, dans le sens où le profil de la catégorie peut être interprété assez facilement. Par contre, l'inconvénient principal de tous les classifieurs linéaires est que l'espace est divisé en seulement deux portions, ce qui peut être restrictif, car tous les problèmes ne sont pas nécessairement linéairement séparables. Les machines à support vectoriel s'apparentent aux classifieurs linéaires, dans le sens où elles tentent de séparer l'espace en deux, mais certaines manipulations mathématiques les rendent adaptables à des problèmes non linéaires. Il y a aussi une famille de classifieurs qui se basent sur l'exemple. On parle alors d'apprentissage à base d'instances. Les nouveaux textes à classer sont comparés directement aux documents de l'ensemble d'entraînement. L'algorithme des k-voisins les plus proches est sans doute le plus connu de cette famille.

2.5.1 Classifieur naïf bayésien

Comme son nom l'indique, ce classifieur se base sur le théorème de Bayes permettant de calculer les probabilités conditionnelles. Ce classifieur cherche la classification qui maximise

la probabilité d'observer les mots du document. Lors de la phase d'entraînement, le classifieur calcule les probabilités qu'un nouveau document appartienne à telle catégorie à partir de la proportion des documents d'entraînement appartenant à cette catégorie. Il calcule aussi la probabilité qu'un mot donné soit présent dans un texte, sachant que ce texte appartient à telle catégorie. Par la suite, quand un nouveau document doit être classé, on calcule les probabilités qu'il appartienne à chacune des catégories à l'aide de la règle de Bayes et des valeurs calculées à l'étape précédente.

La probabilité à estimer est donc : P (c_j |a_1, a_2 , a_3, ..., a_n)

Où :

- c_j est une catégorie
- a_i est un attribut

A l'aide du théorème de Bayes, on obtient $\frac{: P\ (a_1,a_2\ ,a_3,\ldots,a_n|c_j)\ P(c_j)}{P\ (a_1,a_2\ ,a_3,\ldots,a_n)}$

On peut omettre de calculer le dénominateur, qui reste le même pour chaque catégorie.

En guise de simplification, on calcule $P\ (a_1, a_2\ , a_3, \ldots, a_n|c_j)$ ainsi : $\prod_{i=1}^{n}\ P\ (a_i|c_j)$

La probabilité qu'un mot apparaisse dans un texte est indépendante de la présence des autres mots du texte. On sait que cela est faux. Pourtant, cette supposition n'empêche pas un tel classificateur de présenter des résultats satisfaisants. Et surtout, elle réduit de beaucoup les calculs nécessaires. Sans elle, il faudrait tenir compte de toutes les combinaisons possibles de mots dans un texte, ce qui d'une part impliquerait un nombre important de calculs, mais aussi réduirait la qualité statistique de l'estimation, puisque la fréquence d'apparition de chacune des combinaisons serait très inférieure à la fréquence d'apparition des mots pris séparément [20].

Le filtrage bayésien des courriels indésirables (pourriel) est une technique statistique qui s'appuie sur la classification naïve bayésienne pour identifier les messages électroniques non désirés.

Le premier programme de filtrage du courrier électronique utilisant Bayes était le programme iFile de Jason Rennie [19], publié en 1996. Ce programme était utilisé pour classer le courrier en dossiers. La première publication académique sur le filtrage bayésien du pourriel a été faite par Sahami en 1998 [13]. Des variantes de la technique de base ont été implémentées dans plusieurs travaux de recherche et produits logiciels.

SpamBayes est un filtre bayésien anti-spam écrit en Python [8]. La différence la plus notable entre un filtre de bayes classique et le filtre utilisé par SpamBayes, c'est qu'il y a trois catégories au lieu de deux : le spam, non-spam, et incertain. Lors du filtrage d'un message, le

filtre génère un score pour le courrier légitime et un autre pour le spam. Si le score du spam est élevé et le score de légitime est faible, le message sera considéré comme du spam. Si le score de spam est faible et le score de légitime est élevé, le message sera considéré comme légitime. Si les scores sont élevés ou les deux sont faibles, le message sera considéré comme incertain. Cette approche peut entrainer un certain nombre de messages incertains qui ont besoin d'une décision humaine.

2.5.2 Algorithme des k-voisins les plus proches

L'algorithme des k-voisins les plus proches («k-nearest neighbors» ou kNN) est une méthode d'apprentissage à base d'instances. Il ne comporte pas de phase d'apprentissage. Les documents faisant partie de l'ensemble d'entraînement sont seulement stockés. Lorsqu'un nouveau document à classer arrive, il est comparé aux documents d'entraînement à l'aide d'une mesure de similarité [3].

Ses k plus proches voisins sont alors considérés : on observe leur catégorie et celle qui revient le plus parmi les voisins est assignée au document à classer. Une des caractéristiques fondamentales de ce type de classifieur est l'utilisation d'une mesure de similarité entre les documents.

Souvent, on pondère les voisins par la distance qui les sépare du nouveau texte. On accorde plus de poids, lors de la prise de décision, aux documents plus similaires.

Distance euclidienne entre deux documents a et b :

$$\sqrt{\sum_{t \in T} (P_t(a) - P_t(b))^2}$$

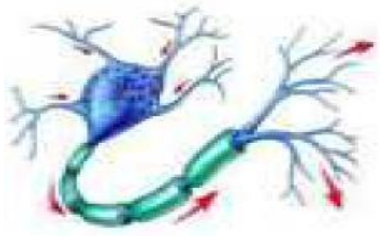
Où :

- T est l'ensemble des attributs ;
- $P_t(a)$ est le poids du terme t dans le document a ;
- $P_t(b)$ est le poids du terme t dans le document b.

Comme il a déjà été mentionné, le classificateur kNN n'implique pas de phase d'entraînement en tant que telle. Cela nous amène à discuter d'une classe d' «apprentisseurs» qu'on peut qualifier de paresseux («lazy learners»). La seule opération préalable est le stockage des exemples d'entraînement. L'apprentissage est repoussé au moment où un nouveau document à classer arrive. À ce moment, il trouve les k documents les plus proches en utilisant une mesure de similarité. Il lui assigne la classe majoritaire, ou pondérée par la distance qui le sépare des

exemples. Par le fait même, la plus grosse part de l'effort requis en termes de temps de calcul est fournie au moment même de la classification.

2.5.3 Réseaux de neurones



Un réseau de neurones (ou Artificial Neural Network) est un modèle de calcul dont la conception est très schématiquement inspiré du fonctionnement de vrais neurones. [5] Les réseaux de neurones sont généralement optimisés par des méthodes d'apprentissage de type statistique grâce à leur capacité de classification et de généralisation, tels que la classification automatique de codes postaux. Ils enrichissent avec un ensemble de paradigmes permettant de générer de vastes espaces fonctionnels, souples et partiellement structurés. Ils appartiennent d'autre part à la famille des méthodes de l'intelligence artificielle qu'ils enrichissent en permettant de prendre des décisions s'appuyant davantage sur la perception que sur le raisonnement logique formel.

Un réseau de neurone **(Fig.2.2)** est en général composé d'une succession de couches dont chacune prend ses entrées sur les sorties de la précédente. Chaque couche (*i*) est composée de N_i neurones, prenant leurs entrées sur les N_{i-1} neurones de la couche précédente. À chaque synapse est associé un poids synaptique, de sorte que les N_{i-1} sont multipliés par ce poids, puis additionnés par les neurones de niveau *i*, ce qui est équivalent à multiplier le vecteur d'entrée par une matrice de transformation [6].

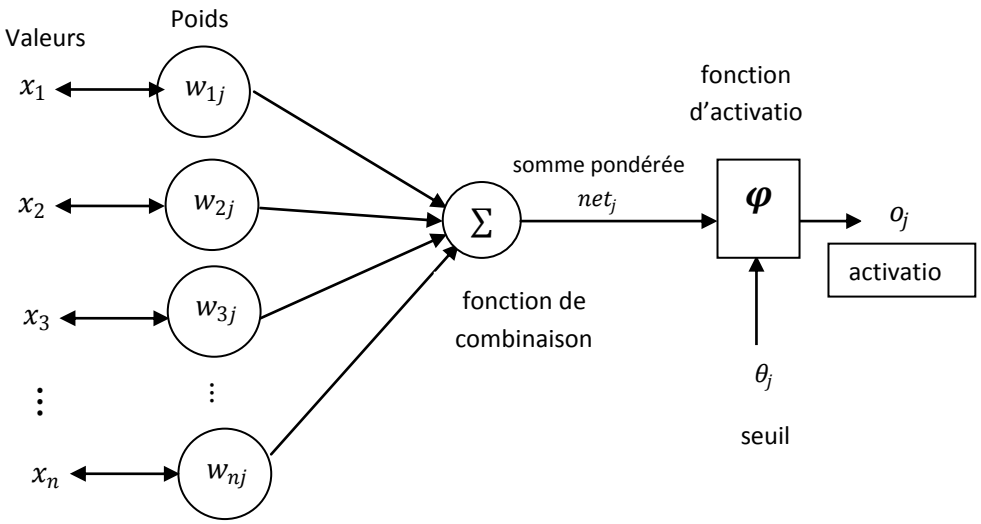


Figure 2.2 - Le modèle fonctionnel des réseaux de neurones [6].

Le réseau de neurones ne fournit pas toujours de règle exploitable par un humain. Le réseau reste souvent une boite noire qui fournit une réponse quand on lui présente une donnée, mais le réseau ne fournit pas de justification facile à interpréter.

Au niveau de filtrage des courriels, SpamAssassin est un filtre installé au niveau du serveur de messagerie qui se base sur les réseaux de neurone. Publié en 2001, est un programme en Perl qui positionne deux nouveaux en-têtes au message : X-Spam-Status et X-Spam-Level. Il fait passer un certain nombre de tests au message. En fonction du résultat de ces tests, il attribue un score au message. Si le score dépasse un certain seuil, le courriel est alors considéré comme du Spam.Ces deux en-têtes permettent alors de créer des boites dans le dossier de messagerie pour orienter le message. L'utilisateur est invité à consulter régulièrement ces boites pour vérifier qu'il n'y a pas eu de faux-positifs [5].

2.5.4 Machines à support vectoriel

Les machines à support vectoriel («Support Vector Machines» ou SVM) [24] forment une classe d'algorithmes d'apprentissage qui peuvent s'appliquer à tout problème qui implique un phénomène f et qui, à partir d'un jeu d'entrées x, produit une sorti $y = f(x)$, et où le but est de retrouver f *à* partir de l'observation d'un certain nombre de couples entrée/sortie. Cette technique tente de séparer linéairement les exemples positifs des exemples négatifs dans l'ensemble des exemples. Chaque exemple doit être représenté par un vecteur de dimension n. Le problème revient à trouver une frontière de décision qui sépare l'espace en deux régions, à trouver l'hyperplan qui classifie correctement les données et qui se trouve le plus loin possible de tous les exemples. On dit qu'on veut maximiser la marge, la marge étant la distance du point le plus proche de l'hyperplan (voir **Figure 2.3**). Intuitivement, cela garantit un bon niveau de généralisation car de nouveaux exemples pourront ne pas être trop similaires à ceux utilisés pour trouver l'hyperplan mais être tout de même situés franchement d'un côté ou l'autre de la frontière [27].

Une propriété intéressante des SVM est que la surface de décision est déterminée uniquement par les points qui en sont les plus près, les vecteurs de support. En présence de ces seuls exemples d'entraînement, la même fonction serait apprise. C'est une différence par rapport à des algorithmes comme kNN avec lesquels tous les exemples d'entraînement sont utilisés lors de l'apprentissage. Cela en fait une méthode très rapide.

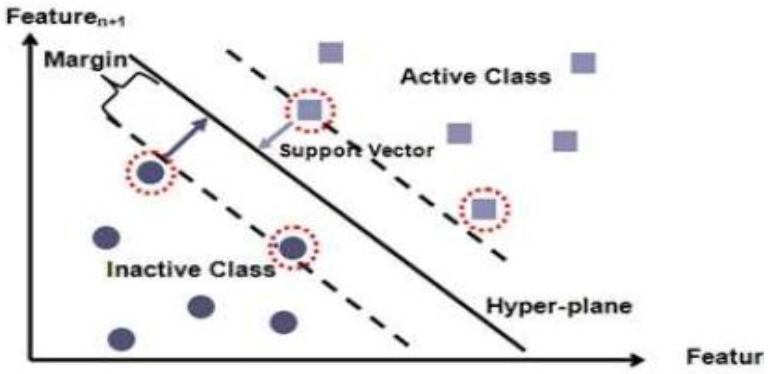


Figure 2.3 - La recherche de la marge maximale [26].

Même s'ils cherchent l'hyperplan séparant l'espace vectoriel en deux, l'avantage des SVM est qu'ils s'adaptent facilement aux problèmes non linéairement séparables. Avant de procéder à l'apprentissage de la meilleure séparation linéaire, on transforme les vecteurs d'entrée en vecteurs de caractéristiques de dimension plus élevée. De cette façon, un séparateur linéaire trouvé par un SVM dans ce nouvel espace vectoriel devient un séparateur non linéaire dans l'espace original. Cette transformation des vecteurs se fait à l'aide de noyaux «kernels» [27].
Dans le cas de la classification de textes, les entrées sont des documents et les sorties sont des catégories. En considérant un classificateur binaire, on voudra lui faire apprendre l'hyperplan qui sépare les documents appartenant à la catégorie et ceux qui n'en font pas partie. Les SVM conviennent bien pour la classification de textes [10]. Une dimension élevée ne les affecte pas puisqu'ils se protègent contre le sur-apprentissage. Dans le même sens, T. Joachim [10] affirme que peu d'attributs sont totalement inutiles à la tâche de classification et que les SVM permettent d'éviter une sélection agressive qui aurait comme résultat une perte d'information. On peut se permettre de conserver plus d'attributs. Également, une caractéristique des documents textuels est que lorsqu'ils sont représentés par des vecteurs, une majorité des entrées sont nulles. Or, les SVM conviennent bien à des vecteurs dits clairsemés. Un autre aspect positif des SVM est qu'aucun ajustement de paramètres manuel n'est requis, car ils ont l'habileté de trouver automatiquement des paramètres adéquats [20].

2.5.5 Les arbres de décision

Un arbre de décision est un outil d'aide à la décision et à l'exploration de données. Il permet de modéliser simplement, graphiquement et rapidement un phénomène mesuré plus ou moins complexe. Sa lisibilité, sa rapidité d'exécution et le peu d'hypothèses nécessaires a priori expliquent sa popularité parmi les méthodes d'apprentissage. Les algorithmes connus sont ID3

[17] et C4.5 [18]. Ils sont également populaires pour la classification de documents.
Les arbres de décision sont construits de la façon suivante :

- A chaque nœud interne correspond un test (un IF...THEN) qui est une question sur l'un des attributs de l'objet.

Chaque test regarde la valeur d'un attribut de chaque exemple. En effet, on suppose qu'un exemple est un ensemble d'attributs/valeurs. Pour des documents, chaque attribut peut être un mot, et la valeur sera par exemple 0 ou 1 selon que ce mot appartient ou non au document.

- A chaque arc correspond une valeur de l'attribut.
- A chaque nœud terminal est associée une classe. Lors de la construction de l'arbre il pourra y avoir plusieurs nœuds terminaux différents de même classe.

Ce sont des classificateurs qui se construisent récursivement. A chaque itération, on cherche l'attribut le plus discriminant, en utilisant un calcul statistique qui détermine dans quelle mesure cet attribut sépare bien les exemples et on l'assigne comme test du nœud. On crée alors un nœud contenant ce test, et on crée autant de descendants que de valeurs possibles pour ce test. Puis on sépare l'ensemble des exemples en plusieurs sous-ensembles suivant la valeur de l'attribut. Et on réitère le processus.
Si l'on doit classer des documents dans des catégories, il faut construire un arbre de décision par catégorie. Les feuilles indiquent la catégorie d'un document avec une certaine probabilité.
Pour déterminer à quelle(s) catégorie(s) appartient un nouveau document, on utilise l'arbre de décision de chaque catégorie auquel on soumet le document à classer. Chaque arbre répond Oui ou Non (il prend une décision).

2.6 Évaluation de la qualité d'une catégorisation

De nombreuses manières d'évaluer la performance du système de catégorisation ont été proposées dans la littérature. De manière générale, on établit d'abord la matrice de confusion à partir de laquelle seront déduites les mesures d'évaluation des performances [28].
La matrice de confusion : il s'agit d'un tableau de contingence qui permet de comparer la classification de l'expert à celle du classificateur. Les lignes représentent le nombre d'occurrence des documents de la classe estimée [classifieur] alors que les colonnes représentent le nombre d'occurrence des documents de la classe réelle [expert].
Pour un système de catégorisation binaire, on a la matrice illustrée dans **le tableau 2.1** : suivant:

Réalité / Système	C	NOT C
C	*Vrai Positif*	*Faux Positif*
NOT C	*Faux Négatif*	*Vrai Négatif*

Le tableau 2.1 La matrice de confusion

Où :

- Vrai Positif (VP) : ratio d'élément de la classe C ayant été étiquetés C par le classificateur.
- Vrai Négatif (VN) : ratio d'élément de la classe nonC ayant été étiquetés nonC par le classificateur.
- Faux Négatif (FN) : ratio d'élément de la classe nonC ayant été étiquetés C par le classificateur.
- Faux Positif (FP) : ratio d'élément de la classe C ayant été étiquetés nonC par le classificateur.

Les différents rapports que l'on peut extraire de la table permettent de définir les critères suivants [28] :

1. **La précision (Precision)** : est définie par le nombre d'assignations correctement produites par rapport au nombre d'assignations produites par le système.

$$P = \frac{\mathrm{VP}}{(\mathrm{VP} + \mathrm{FP})}$$

2. **Le rappel (Recall)** : est défini par le nombre d'assignations produites correctement par le système par rapport au nombre d'assignations produits par les experts humains.

$$R = \frac{\mathrm{VP}}{(\mathrm{VP} + \mathrm{FN})}$$

Ces deux mesures sont antagonistes : avec un même système, augmenter la précision se fait au détriment du rappel, et inversement. On cherche bien sur à atteindre les performances du système idéal pour lequel : $R = P = 1$. Si l'on veut comparer les performances de différents systèmes, il faut donc prendre en compte le rappel et la précision. Cependant, il n'est pas aisé de comparer des couples de valeurs. La F-mesure «break-even point» est une manière de combiner ces deux valeurs, c'est le point où la précision et le rappel sont égaux. Plus ce point

se rapproche de 100%, plus le classificateur est performant à la fois en précision et en rappel comme suit :

$$F = \frac{(\beta^2+1)RP}{(R+P)}$$

Différentes valeurs de β donnent différents critères pour lesquels l'importance du rappel et de la précision varie. Dans la littérature, les résultats sont souvent exposés avec β = 1.

On appelle alors cette mesure la *F*1– *mesure* [29].

3**. L'erreur (Error)** : représente la proportion des documents mal classés.

$$Err = \frac{(FP+FN)}{(VP+VN+FP+FN)}$$

4. **L'exactitude (Accuracy)** : représente la proportion des documents bien classés.

$$Acc = \frac{(VP+VN)}{(VP+VN+FP+FN)}$$

Conclusion

Nous avons passé en revue, dans ce chapitre la notion de catégorisation de documents. Nous avons défini ses principales phases, en précisant les principaux modes de représentation des documents utilisés et les principales techniques appliquées pour réduire la taille du vocabulaire pris en compte. Nous avons réalisé une étude de différentes méthodes d'apprentissage automatique.

Cette étude nous a permis de mieux connaître les caractéristiques de ces méthodes et leur comportement sur les données. Enfin, nous avons présenté les critères d'efficacité pertinents les plus courants afin d'évaluer un classificateur de document.

Comme nous venons de le voir dans ce chapitre la catégorisation de documents nécessite l'utilisation d'un algorithme d'apprentissage supervisé pour affecter à un document la bonne catégorie. Le prochain chapitre détaille des méthodes de filtrage des courriels que nous avons comparé.

Chapitre 3

Les méthodes choisies

3.1 Les classifieurs avec les spam

Un classifieur linéaire[10] est un type d'algorithme en intelligence artificielle. La fonction d'un classifieur est de trier un ensemble d'échantillons en différentes classes selon les propriétés des ces éléments. Un classifieur linéaire est un type particulier de classifieur qui se base sur une combinaison linéaire des différentes propriétés.

- **Un ensemble des caractéristiques** : La base de données
- ***Deux classes*** : le spam et le ham.
- ***Une tétra-pelletée de propriétés*** : le résultat des filtres
- ***Des poids associés aux propriétés*** : le score associé à chaque filtre

  Schématiquement, avec **S** pour Spam et **H** pour Ham et en prenant la projection sur un espace à deux dimensions (i.e. Ne considérant que deux propriétés/filtres) :

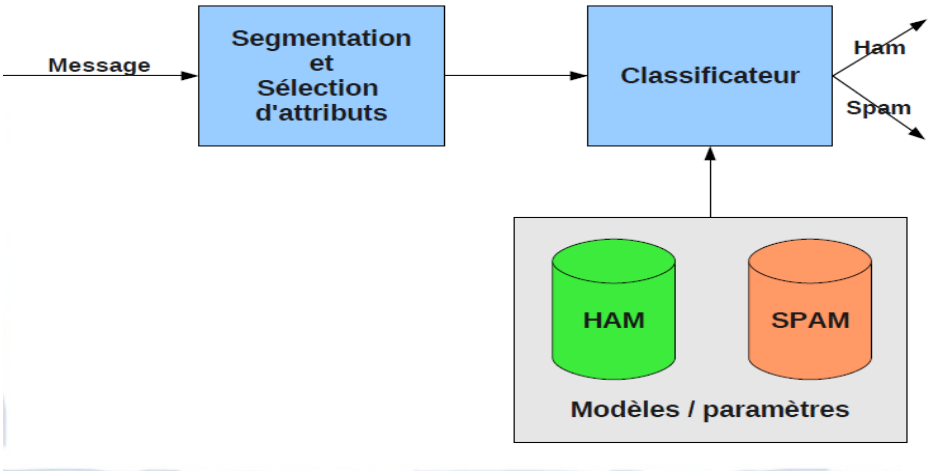


Figure 3.1 – Le processus de classificateur avec les SPAM [33].

Le classifieur doit être pertinent ce qui veut dire :

- détecter le plus de spam possible
- éviter au maximum les faux positifs (c'est-à-dire éviter de considérer un ham comme étant un spam)

Les mathématiques proposent différentes théories et modèles de calcul qui peuvent être appliqués à cette problématique de classification.

[10] C:http ::// Classifieurs linéaires, SVM SAMIFiS Système d'Analyse Multicritère Intelligent pour Filtrage de Spam.html

Dans le cadre de nos expérimentations, nous avons choisi d'utiliser les méthodes d'apprentissage automatique pour la classification : les Support Vector Machines (SVM), le Naïve Bayes (NB) et les arbres de décisions (C4.5). On a déjà introduit ; les (3) méthodes choisies pour la comparaison (SVM, NB, C4.5) sont bien détaillées au **chapitre 2**.

3.2. Outils d'analyse

3.2.1 Le logiciel libre de datamining

Le logiciel libre permet aux chercheurs un développement sans contraintes et aux utilisateurs une utilisation sans mauvaises surprises [30].

Un logiciel libre de datamining permet : La diffusion du logiciel ; il est accessible à tous ce qui permet de faire la comparaison et la vérification des résultats ; reproduire « exactement » les expérimentations. D'autre part, il permet de comparer les interprétations d'un même problème (ex. Relieff WEKA) et lecture du code par d'autres chercheurs (ex. Naive Bayes classifier). Par ailleurs le logiciel libre montre et partage des bibliothèques (ex. générateurs aléatoires, fonctions de répartitions, les fameux packages...).

- Les critères des logiciels commerciaux éludés [30] :

  1. Interfaçage avec les bases de données
  2. Traitement à la volée sur de très grandes bases de données (pas de données en mémoire)
  3. Déploiement des modèles construits et mis en production
  4. Reporting évolué et dynamique (m. à j. à la volée dans les documents édités)
  5. Exploration graphique évoluée et interactive (isoler graphiquement des sous-populations)

2.2 Exemple de logiciels libres :

Des outils riches mais des objectifs, des cultures et des approches très différentes [31] :

SIPINA : Il est basé sur le pilotage par menu, il est simple au premier abord mais ingérable dès que le logiciel gagne en complexité comme il est impossible de garder la trace d'une analyse complète et donc de la reproduire. D'autre part il exige une documentation complète et constamment à jour (Open Stat & Stat 4U sont dans la même situation)

R : Il utilise le langage de programmation avec toute la puissance d'un langage de programmation. L'accès au langage est une barrière à l'entrée qui rebute certains

KNIME : Il est basé sur les diagrammes de traitements, avec une autre manière de présentation diagrammes de traitements. KNIME est un des très rares à savoir représenter une boucle (notion de « méta composant»).

ORANGE : Un outil très marqué «machine learning», souple avec des efforts pour la convivialité et la simplicité[11].

WEKA: Possède une référence mondiale, une bibliothèque de méthodes très riche. Il est orienté machine learning. Il permet l'ajout de méthodes, l'ajout de classes et la recompilation. Il se caractérise par un accès compliqué à l'interface et possède l'atout de pouvoir effectuer un appel des classes en externe [31].

TANAGRA : Tanagra[12] est un logiciel gratuit d'exploration de données destiné à l'enseignement et à la recherche. Il implémente une série de méthodes de fouille de données issues du domaine de la statistique exploratoire, de l'analyse de données, de l'apprentissage automatique et des bases de données. C'est un projet ouvert au sens qu'il est possible à tout chercheur d'accéder au code, d'ajouter ses propres algorithmes et de diffuser, toujours gratuitement, le logiciel modifié. Tanagra est diffusé depuis décembre 2003. Le logiciel a été réalisé pour un environnement WIN32. Il s'exécute donc sous Windows, ou sous Linux via WINE1.

Après avoir effectué une prospection des logiciels libres disponibles notre choix s'est fixé sur le logiciel Tanagra pour sa facilité d'utilisation et sa convivialité ainsi que sa popularité.

[11] http://eric.univ-lyon2.fr/~

[12] http://fr.wikpidia.org/wiki/tanagra

Chapitre 4

Expérimentations

4.1 La description des données:

Pour effectuer notre travail, nous avons utilisé la collection SpamBase[13] disponible dans le Machine Learning Repository de l'UCI (University of California Irvin). Elle contient les informations relatives à 4601 messages, avec 1813 (39,4%) de spam. Cette collection a été prétraitée, et les textes des messages ne sont pas disponibles. Chaque message est décrit à l'aide de 57 attributs, le 58ème correspond à la classe du message (spam ou non). Les caractéristiques (55-57) mesurent la longueur des séquences de lettres majuscules consécutives. Les définitions des attributs de SpamBase est comme suit :

- 48 attributs représentent un ensemble de mots (Ex : adress, all, conference, telnet, data, meeting,…).
- 6 attributs représentent un ensemble de caractères [‘ ;’ ,’(‘ ,’[‘ ,’ !’ ,’$’ ,’#’].
- 3 attributs mesurent la longueur moyenne, minimale et maximale des séquences ininterrompues de lettres majuscules.
- 1 attribut nominal qui indique si le courriel a été considéré comme spam ou non.

Cette description est résumée dans Le Tableau 4.1.

Nombre d’attributs	Type	Description
48	réel [0,100]	Fréquence de l’attribut (mot).
6	réel [0,100]	Fréquence de l’attribut (caractère).
1	réel [1,…]	Longueur moyenne des séquences ininterrompues de lettres majuscules.
1	entier [1,…]	Longueur des plus longues ininterrompues séquences de lettres majuscules.
1	entier [1,…]	somme de la longueur des séquences ininterrompues de lettres majuscules.
1	[0,1]	Le (1) indique si le courriel a été considéré comme spam ; sinon (0)

Le tableau 4.1 La description des attributs

4.2 Résultats : Evaluation des performances de classifieur

Pour comparer les trois méthodes (SVM, Naïve Bayes et C4.5) de classification, nous nous sommes limités à la phase d'apprentissage. Pour mesurer l'importance des attributs utilisés et

[13] http://archive.ics.uci.edu/ml/datasets

comparer la performance des trois méthodes choisies, nous allons étudier la performance de chacune des méthodes en fonction des attributs sélectionnés. La performance est mesurée en termes d'erreur d'apprentissage BER (Balanced Error Rate) qui représente la proportion des documents mal classés dont nous rappelons la formule:

$$Err = \frac{(FP+FN)}{(VP+VN+FP+FN)}$$

Dans notre cas, un résultat positif est un message spam et un résultat négatif est un message légitime. Les cas étudiés sont définis comme suit:

Cas 1 : Tous les 57 attributs sont utilisés.

Cas 2 : Seulement les 48 attributs relatifs aux fréquences des mots clés sont utilisés.

Cas 3 : Seulement les 6 attributs relatifs aux fréquences des caractères clés sont utilisés.

Cas 4 : Seulement les 3 attributs relatifs aux longueurs des séquences ininterrompues de lettres capitales sont utilisés.

Cas 5 : Les 48 attributs relatifs aux fréquences des mots clés et les 6 attributs relatifs aux fréquences des caractères clés sont utilisés.

Cas 6 : Les 48 attributs relatifs aux fréquences des mots clés les 3 attributs relatifs aux longueurs des séquences ininterrompues de lettres capitales sont utilisés.

Cas 7 : Les 6 attributs relatifs aux fréquences des caractères clés et les 3 attributs relatifs aux longueurs des séquences ininterrompues de lettres capitales sont utilisés.

Les résultats obtenus sur la base d'apprentissage avec les trois méthodes testées sont résumés dans le tableau 4.2. D'autre part Figure 4.1 donne une représentation graphique de ces résultats.

Les méthodes choisies	***C4.5***	***SVM***	***NB***
Cas 1	0.0367	0.0924	0.1126
Cas 2	0.0478	0.1106	0.1274
Cas 3	0.1095	0.2997	0.2745
Cas 4	0.1162	0.3588	0.3388
Cas 5	0.0422	0.0982	0.1171
Cas 6	0.0361	0.1035	0.1193
Cas 7	0.0689	0.2639	0.2541

Tableau 4.2 – Erreur d'apprentissage obtenues avec les 3 méthodes choisies pour les 7 cas étudiés.

4.3 Discussion des résultats

Nous pouvons constater dans le Tableau 4.2 et la Figure 4.1 que les classifieurs SVM et Naïve Bayes ont des performances relativement proches. Par contre C4.5 est meilleur que les deux autres, il est capable de diminuer l'erreur de classification d'une façon remarquable. La méthode C4.5 réalise une erreur de 0.03 contre une erreur autour de 0.1 pour les deux autres méthodes dans les cas 1, 2, 5 et 6. De même la méthode C4.5 atteint une erreur qui varie de 0.04 à 0.07 contre une erreur autour de 0.3 pour les deux autres méthodes dans les cas 3, 4 et 7. Ces résultats vont dans le même sens que ceux exposés dans [32] qui contient une comparaison des deux méthodes SVM, NB avec la sélection des attributs.

Nous constatons dans le tableau 4 .1 que les attributs de type fréquences des mots clés ont une influence importante sur l'erreur d'apprentissage. Par exemple pour les cas 1, 2, 5 et 6 avec C4.5 l'erreur d'apprentissage est autour de 0.04 alors que pour les cas 3 et 4, l'erreur passe à une valeur de 0.1.

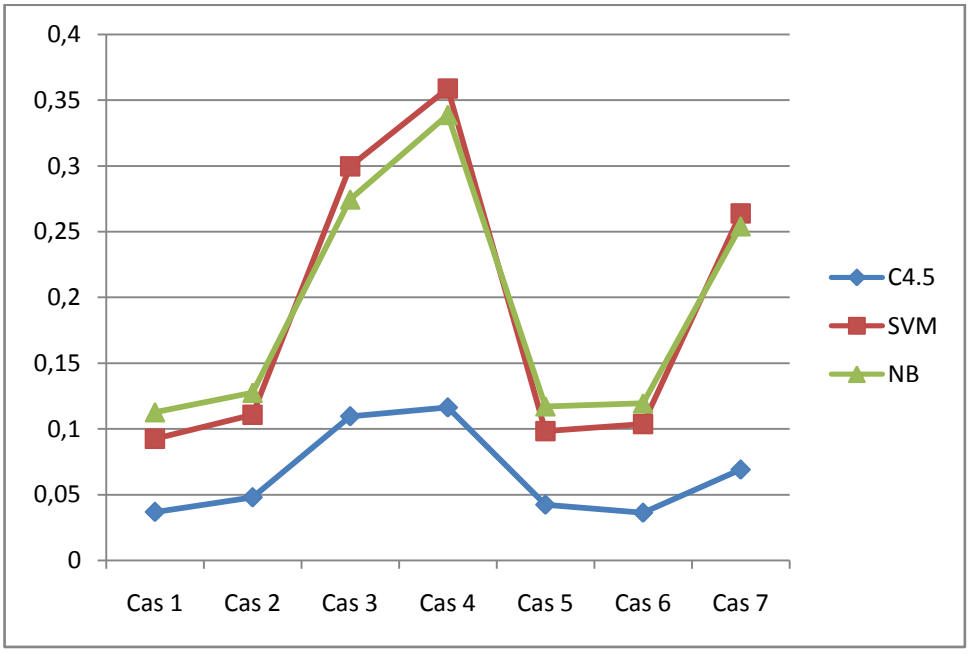


Figure 4.1 - Erreur d'apprentissage de chacun des classifieurs SVM, NB et C4.5 pour les sept cas étudiés.

Conclusion générale

La sélection des attributs est une étape très importante dans une procédure de classification. L'objectif étant de réduire le nombre de caractéristiques utilisées, tout en essayant de maintenir ou d'améliorer les performances de la classification.

Dans ce projet nous avons comparé les méthodes de classifications, basé sur les textes des messages considérés, dans le but de réduire le nombre des caractéristiques utilisées et d'améliorer les performances de ce filtre. Le travail effectué nous a permis de comparer trois de ces méthodes (SVM, Naïve Bayes et C4.5) dans le cadre de la détection de spams, donnant ainsi quelques indications sur celles qu'il serait intéressant d'utiliser dans une application de ce type.

Bien que les résultats obtenus soient intéressants et encourageants, beaucoup de points sont susceptibles d'être étudiés dans le cadre de travaux futurs, tels que l'utilisation d'autres mesures de sélection dans l'étape d'évaluation individuelle des caractéristiques, l'utilisation d'autres classifieurs dans l'étape de validation de sous ensembles de caractéristiques, et l'utilisation d'autres ensembles d'attributs, en mettant en œuvre les étapes de prétraitement textuels et d'extraction de caractéristiques sur des bases de données de textes de messages électroniques.

Références

[1] Kjersti Aas and Line Eikvil. Text categorization : a survey. Technical report, Norwegian Computing Center, 1999.

[2] Ferris Research Postini CNIL Basex, Radicati Group. *Anti-Spam technologie*. In : Site de la Vade-Retro [en ligne], 2010.

[3] R. Geetha Ramani, G. Sivagami, Parkinson Disease Classification using Data Mining Algorithms, International Journal of Computer Applications (0975 – 8887) Volume 32– No.9, October 2011.

[4] Sebastiani (F.) Caropreso (M.), Matwin (S.). A learner-independent evaluation of the use fulness of statistical phrases for automated text categorization, pages 78-102, 2001.

[5] Chris Miller. Neural network-based antispam heuristics. In Symantec, white paper, 2003.

[6] Dagan, I., Karov, Y., & Roth, D. 1997. Mistakedriven learning in text categorization. 2nd Conference on Empirical Methods in Natural Language Processing, EMNLP-1997,

[7] Bruno Rasle Frédéric Aoun. Halte au spam, volume 294 of ISBN 2212113072, 9782212113075. , Paris, Eyrolles, 2003.

[8] Paul Graham. A plan for spam. Introduction aux modèles graphiques, 7, 2002.

[9] Radwan Jalam. Apprentissage automatique et catégorisation de textes multilingues. PhD thesis, Université Lumière Lyon2, 2003.

[10] T. Joachims. Text categorization with support vector machines: learning with many relevant features. In Proceeding of ECML-98, 10th European Conference on Machine Learning.

[11] Jean-Louis Laurière. Intelligence artificielle, chapter Tome 1 : Résolution de problèmes. Eyrolles, 1986.

[12] Julie Beth Lovins. Development of a stemming algorithm. Technical report, Mechanical Translation and Computational Linguistics, 1968.

[13] D. Heckerman M. Sahami, S. Dumais and E. Horvitz. A bayesian approach to filtering junk e-mail. In Learning for Text Categorization Papers from the AAAI Workshop, Madison Wisconsin.

[14] M. Abdenour Mokrane. Représentation de collections de documents textuels : application à la caractérisation thématique. PhD thesis, Université de Montpellier II, France, Novembre 2006.

[15] CNIL : Commission nationale de l'informatique et des libertés. Spam :Défi nition, 2008.

[16] M.F. Porter. An algorithm for suffix stripping. 14 :130-137, 1980.

[17] J. R. Quinlan. Induction of decision trees. Machine Learning, 1(1) :81-106.

[18] J. R. Quinlan. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers Inc., 1993.

[19] Jason Rennie. iFile : An application of machine learning to e-mail filtering. Technical report, 1996.

[20] Simon Réhel. Catégorisation automatique de textes et cooccurrence de mots provenant de documents non étiquetés. Master's thesis, Faculté des études supérieures de l'Université Laval, 2005.

[21] M. Roche and Y. Kodrato®. Utilisation de LSA comme première étape pour la classification des termes d'un corpus spécialisé. In Actes (CD-ROM) de la conférence MAJECSTIC' 03 (Manifestation des Jeunes Chercheurs dans le domaine STIC), 2003.

[22] Gerard Salton and Christopher Buckley. Term weighting approaches in automatic text retrieval. Information Processing & Management, 1987.

[23] George W.Furnas Thomas K. Landauer Scott Deerwester, T.Susan Dumais and Richard Harshman. Indexing by latent semantic analysis. American Society of Information Science, 1990.

[24] Vladimir.N Vapnik. The nature of statistical learning theory. Springer Verlag, 1995.

[25] R. Kishore Kumar, G. Poonkuzhali, P. Sudhakar, Comparative Study on Email Spam Classifier using Data Mining Techniques ,Marsh 14-16 2012

[26] Mohamadally. H, Fomani Boris. SVM : Machines a Vecteurs de Support ou Separateurs a Vastes Marges, BD Web, ISTY3 Versailles St Quentin, France, janvier 2006

[27] Drucker, H., Vapnik, V., & Wu, D. 1999.Automatic text categorization and its applications to text retrieval. IEEE Trans. 10, 1048–1054.

[28] Yang, Y.An evaluation of statistical approaches to text categorization. Inform. Retr., 1, 69–90, 1999.

[29] Rijsbergen, C. J. V. Information Retrieval. Butterworths. 1979.

[30] Ricco RAKOTOMALALA, Les logiciels gratuits pour l'enseignement du Data Mining.

[31] Ricco RAKOTOMALALA .Tanagra Un logiciel de Data Mining gratuit pour l'enseignement et la recherche, Mars 2010.

[32] Kamilia MENGHOUR, Labiba SOUICI-MESLATI. Sélection de Caractéristiques pour le Filtrage de Spams, Laboratoire LRI, Université Badji Mokhtar, Annaba, Algérie.

[33] Jose-Marcio. Filtres statistiques de spam : Etat de l'art et utilisation collective, OSSIR – 07 juillet 2009.