الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Centre Universitaire Salhi Ahmed – NAAMA
Institut des Sciences et de Technologie
Département de Mathématiques et Informatique

MEMOIRE

En vue de l'obtention du diplôme de MASTER Académique

En : INFORMATIQUE

Spécialité : Systèmes d'informations

Présenté Par : BESSAD Riyadh
CHAALA Zakarya

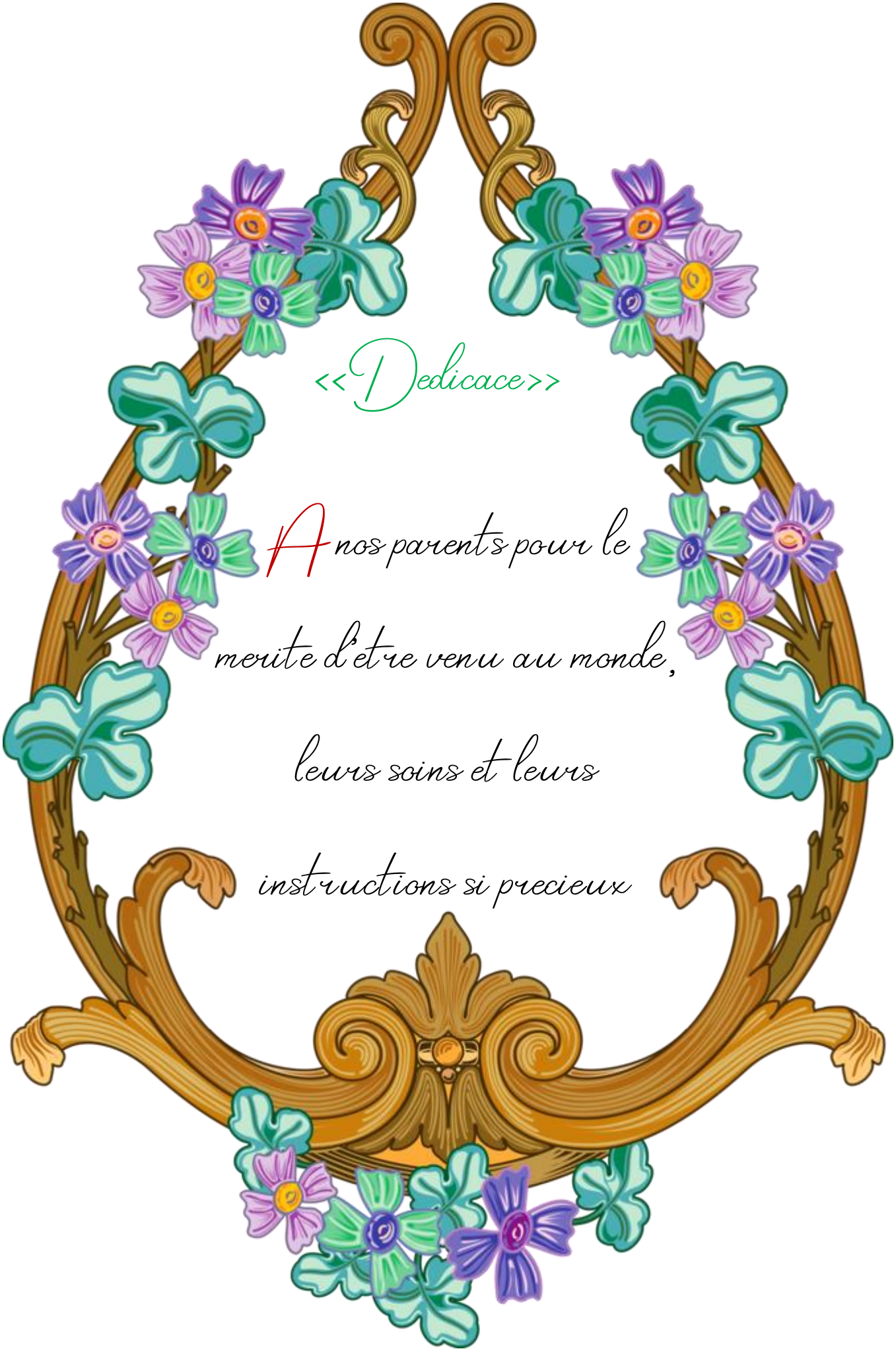
Intitulé

DETECTION DES SPAM:
Une approche à base d'apprentissage automatique (SVM)

Soutenu, devant le jury composé de :

Président	SIDAOUI Boutkhil
Encadreur	BOUOUGADA Bennamar
Examinatrice	ABDELMOUMEN Halima
Examinateur	KAOUAN Moussa

Session : (Mois juillet 2021)
Promotion : 2020 / 2021



<<Dedicace>>

A nos parents pour le

merite d'etre venu au monde,

leurs soins et leurs

instructions si precieux

<<Remerciement>>

Avant toute chose nous remercions Allah le tout puissant de nous avoir accorde la force et les moyens afin de pouvoir realiser ce travail.

Nous voudrons tout d'abord adresser toute notre gratitude a notre encadreur, M. Bouougada Bennamar, pour sa patience, sa disponibilite et surtout ses judicieux conseils, qui ont contribue a alimenter notre reflexion.

Nos plus vifs remerciements s'adressent aussi a tous les enseignants de departement de mathematiques et informatique qui nous ont fourni les outils necessaires a la reussite de nos etudes universitaires qu'ils puissent trouver dans ce travail le temoignage de notre sincere gratitude et notre profond respect. Nous tenons a remercier sincerement les membres du jury qui nous font le grand honneur d'evaluer ce travail. Enfin, du point de vue personnel, nos chaleureux remerciements vont a tous nos camarades qui nous ont apporte leur support moral et intellectuel tout au long les annees universitaires.

Bessad Riyadh et Chaala Zaparya

Sommaires

SOMMAIRES i

MOTS-CLES iii

RESUME iv

ABSTRACT v

ABREVIATIONS vi

LISTE DES FIGURES vii

LISTE DES TABLEAUX viii

INTRODUCTION GENERALE 1

CHAPITRE I: SPAM 3

INTRODUCTION 3
1.ORIGINE DU MOT SPAM 3
2.DEFINITION DU SPAM 3
3.NAISSANCE ET DEBUTS DU SPAM 4
4.EVOLUTION DU SPAM 4
5.CATEGORIES DE SPAM 5
Adresses email devinées 5
Récolteur d'adresses 5
Jeux concours 6
Vendeur d'adresses 6
6.HISTORIQUE DE SPAM 6
6.1.Premier envoi massif 6
6.2.Utilisations abusives à fins commerciales 7
5.3.Evolution du spam 7
CONCLUSION 9

CHAPITRE II. ETAT DE L'ART SUR LA DETECTION DES SPAMS 10

INTRODUCTION 10
1. UTILISATION D'UN CLASSIFIEUR DE SPAM HYBRIDE NB-SVM 10
2.DETECTION DE FAUX AVIS A L'AIDE DE L'EXPLORATION DE DONNEES 11
3.MACHINE LEARNING POUR LA DETECTION DES COURRIERS INDESIRABLES 12
4.UN MODELE DE DETECTION DE SPAM INTELLIGENT BASE SUR SYSTEME 13
5.UN SPAM A VIE MODELE DE CLASSIFICATION 13
6.DETECTION EFFICACE DU SPAM D'OPINION 14
7.APPROCHE DE DETECTION DE SPAM POUR UN MESSAGE MOBILE 15
8.DETECTING WEB SPAM BASED ON NOVEL FEATURES 16
CONCLUSION 18

CHAPITRE III L'APPRENTISSAGE AUTOMATIQUE 19

1.DIFFERENTS TYPES D'APPRENTISSAGE 19
1.1.Apprentissage supervisé 19
1.2.Apprentissage non supervisé 19
1.3.Apprentissage semi supervisé 20
2.LES DIFFERENTS METHODES DE APPRENTISSAGE SUPERVISE 20
2.1.Les k plus proches voisins 20
2.2.Réseaux de neurones 22

2.3.Arbres de décision 24
2.4.Support Vector Machines 26
CONCLUSION 28
CHAPITRE IV. SUPPORT VECTOR MACHINES 29
INTRODUCTION 29
1.NOTIONS DE BASES 29
1.1.Hyperplan 29
1.2.Marge 31
2.THEORIE D'APPRENTISSAGE DE VAPNIK-CHERVONENKIS 31
3.PRINCIPE D'UN SVM 34
3.1.Principes fondamentaux 34
3.2.Fondement mathématique 36
3.3.Le choix des paramètres optimaux 47
4.EXTENSION DES SVM 47
4.1.Approche Un-contre-Tous (1vsR) 47
4.2.Approche Un-contre-Un (1vs1) 49
CONCLUSION 51
CHAPITRE V. CONCEPTION ET IMPLEMENTATION DE L'APP EN PYTHON 52
INTRODUCTION 52
CLASSIFICATION DES POURRIELS 53
1- Prétraitement des emails 53
2- Extraction de fonctionnalités à partir d'emails 56
3- Formation SVM pour la classification des spams 57
4- Tester la classification des spams 58
5- Principaux prédicteurs de spam 58
6-Essayez nos propres emails 59
CONCLUSION GENERALE 62
REFERENCE 63

Mots-clés

Spam; SVM ; Vector; Dataset; Pourriel; Ham; Anti-spam; Detection des spams.

Résumé

La plupart des systèmes de filtrage des emails existants enregistrent des faiblesses sur l'efficacité du filtrage. Certains systèmes sont basés seulement sur le traitement de la partie structurée (un ensemble de règles sur l'entête du message), et d'autres sont basés sur un balayage superficiel de la partie texte du message (occurrence d'un ensemble de mots clés décrivant les intérêts de l'utilisateur).

On propose une double amélioration de ces systèmes. D'une part, nous proposons un ensemble de critères automatisables et susceptibles d'influer sur le processus de filtrage. Ces critères sont des indices qui portent généralement sur la structure et le contenu des messages. D'autre part, nous utilisons une méthode d'apprentissage automatique permettant au système d'apprendre à partir de données et de s'adapter à la nature des mails dans le temps. Nous nous intéressons à un type de messages bien particulier, qui continue à polluer nos boîtes emails de façon croissante : les messages indésirables, appelés *spam*. Nous présentons à la fin les résultats d'une expérience d'évaluation.

Abstract

Most of existing filtering messages systems exhibit weaknesses in term of efficiency. In fact, there are systems that use only message header information and others use a superficial processing of message body so we try to improve the filtering processes efficiency. First, we introduce a set of criteria which are cues related to the message structure and content. Second, we use a machine learning method allowing the system to learn from data and to adapt to the email nature. We are interested in a special type of messages that continuously populate our email boxes: *spam* email. At the end, to measure the approach performances, we illustrate and discuss the results obtained by experimental evaluations.

Abréviations

CNIL	La Commission Nationale de l'Informatique et des Libertés.
DEC	Digital Equipment Corporation.
NB-SVM	Naive Bayes - Support Vector Machine
ELCADP	Ensuring lifelong spam classification model using Adjustable Dataset Partitioning.
LR	The binary logistic regression model.
MLP	Multi-layer Perceptron
Knn	The k-nearest neighbors
RNN	Recurrent neural network
MRE	Minimisation de Risque Empirique.
VC	Dimention de Vapnik et Chervonenkis.

Liste des figures

Figure 1 : Evolution du spam 5
Figure 2 : Premier envoi massif. 6
Figure 3 : Architecture NB-SVM 10
Figure 4 : Les k plus proches voisins. 20
Figure 5 : Neurone formel. 22
Figure 6 : Perceptron multicouche 23
Figure 7 : Ordonnancement de trois entiers par Arbre de décision. 25
Figure 8 : Support vectors. 30
Figure 9 : Support vectors. avec les vecteurs de support. 30
Figure 10 : Représentation de la marge 31
Figure 11 : Sous apprentissage 32
Figure 12 : Apprentissage par cœur 33
Figure 13 : Comportement du risque empirique. 33
Figure 14 : Meilleur hyperplan séparateur 34
Figure 15 : Exemple d'un cas linéairement séparable 35
Figure 16 : Exemple de projection dans un espace de redescription 36
Figure 17 : Marge souple et variable élastique εi 42
Figure 18 : Représentation du compromis entre la tolérance C et la variable élastique εi. 42
Figure 19 : Chaîne de traitements génériques d'une méthode à noyau. 46
Figure 20 : Approche un contre tous. 48
Figure 21 : Approche un contre un. 50

Liste des tableaux

Tableau 1 : exemple des email depuis enron tester avec l'application. 61

INTRODUCTION

GENERALE

INTRODUCTION GENERALE

Le courrier électronique (ou courriel, email) est le service le plus utilisé sur internet, il est sans doute la technique qui a changé nos habitudes à une grande échelle. La croissance de l'Internet est reliés directement à l'importance du courriel, car plusieurs sites web lui sont maintenant consacrés, et presque tous les gens qui ont accès à internet ont au moins une adresse de courrier électronique qu'ils vérifient quotidiennement, ce qui explique les milliards des courriels qui s'envoient et sont reçus chaque jour.

Aujourd'hui, Si nous comparons le courrier électronique aux autres moyens de communication, (par écrit, téléphone), nous nous apercevons que les avantages des courriels surpassent ses inconvénients. Sa force réside dans le médium du transport des messages, la rapidité avec laquelle circulent les courriels, l'économie, la disponibilité en tout temps indépendamment du décalage horaire et à la possibilité de les envoyer à plusieurs personnes en même temps. La nature informatique de ces courriels offre des avantages incomparables, dont l'envoi des documents électroniques par attachement, l'archivage des messages est beaucoup plus facile à effectuer qu'avec les communications écrites ou par téléphone, ainsi que, le courrier électronique permet d'effectuer un traitement rapide, efficace et automatique sur les messages comme la recherche par mots clés, le tri automatique par sujet.

Cependant, les utilisateurs se retrouvent assez vite submergés de quantités de courriers électroniques indésirables ou non sollicités appelés aussi spam. En effet, le spam est rapidement devenu un problème majeur sur Internet.

Le spam est un phénomène mondial et massif. Selon la CNIL[1], le spam est défini de la manière suivante: Le "spamming" est l'envoi massif de courriers électroniques non sollicités, à des personnes avec lesquelles l'expéditeur n'a jamais eu de contact et dont il a capté l'adresse électronique de façon irrégulière, Il existe de nombreuses techniques contre le spam qui peuvent être divisées en deux groupes. Le 1^{ier} contient les solutions basées sur l'entête du message électronique telles que les listes noires et les listes blanches. Le 2ème groupe de solutions contient celles qui sont basées sur le contenu textuel du message telles que la détection basé sur l'apprentissage automatique.

[1] La Commission Nationale de l'Informatique et des Libertés.

Dans notre projet de fin d'études nous allons présenter une méthode basée sur l'apprentissage automatique.

Nous avons décomposé notre mémoire en cinq chapitres. Le premier chapitre vise à présenter la définition de spam, à travers ses objectifs, ses contenus et ses impacts, son évolution et ces catégories le deuxième chapitre expose quelques travaux publiés sur la détection des spams, le troisième chapitre Présenter l'apprentissage automatique de la manière la plus simple possible, le quatrième chapitre on parle les machines à vecteurs de support (SVM) et Le cinquième chapitre on va fini par la conception et l'implémentation de l'application de classification des emails (spam ou non-spam)

Chapitre 1:

Spam

Chapitre I: Spam

Introduction

Le spam est un grand problème pour les internautes. Les augmentations récentes du taux de spam ont causé une grande inquiétude parmi la communauté Internet. De nombreuses solutions avaient été suggérées pour résoudre le problème.

Dans ce chapitre, nous présentons tout d'abord les débuts du spam, ses objectifs, ses contenus, ses impacts et les différentes techniques utilisées pour détecter ce type de courriels.

1.Origine du mot spam

En 1937 La société Hormel Foods organise un concours pour trouver un nouveau nom pour leur jambon épicé, Ce nom doit être aussi caractéristique que le goût du produit «Spiced Ham» et qui propose « Spam » pour ce produit, fut donc la marque retenue.

Cette viande précuite en boîte souvent synonyme de mauvaise nourriture a été largement utilisée par l'intendance des forces armées américaines pour la nourriture des soldats pendant la Seconde Guerre mondiale et sera introduite dans diverses régions du monde à cette occasion. [1]

2.Définition du spam

Le spam est un message électronique non sollicité, envoyé massivement à un grand nombre de destinataires, à des fins publicitaires ou malveillantes.

Le terme spam est aussi utilisé pour désigner le même type de message transmis par d'autres moyens de communication électroniques tels que les messageries instantanées, les blogs, les forums, et plus récemment, les réseaux de téléphonie mobile, via les SMS ou MMS. Même si le moyen de communication est différent, les techniques d'envoi et de détection restent relativement similaires.

Sur le réseau ARPANET, dans la date du 3 mai 1978. Ce jour-là c'est le jour du premier spam, Gary Thuerk, commercial de la société informatique DEC3, invitait par email 393 personnes à découvrir sa nouvelle machine, le 2020.

3.Naissance et débuts du spam

Le phénomène du spam, qui se manifestait au début par des messages publicitaires pour des produits ou services commerciaux, a connu une croissance exponentielle, atteignant son point d'inflexion en 2004.

En outre, les simples messages publicitaires ont été remplacés par des messages potentiellement dangereux, et non plus seulement importuns. Le spam est aujourd'hui de nature trompeuse, il peut perturber le fonctionnement du réseau et sert de vecteur de propagation des virus, ce qui sape la confiance des consommateurs, laquelle est un préalable indispensable à la société de l'information et au succès du commerce électronique. [1]

4.Evolution du spam

Cette évolution (voir la Figure 1) peut se résumer en trois grands changements :

•Premièrement, les spammeurs ont adopté de nouvelles méthodes techniques et sociales pour dissimuler l'origine des messages qu'ils envoient, ce qui leur permet de contourner les mesures prises à leur encontre par les autorités chargées de l'application des lois, les FAI et les internautes. Ils ont ainsi recours notamment à la falsification de courriels, à l'utilisation de relais et de serveurs mandataires ouverts et, de plus en plus, à des réseaux d'ordinateurs zombies[2], ou botnets[3].

•Deuxièmement, ces derniers mois, le spam est devenu un vecteur pour la propagation de diverses menaces, facilitant la diffusion de virus et d'autres logiciels malveillants ou servant de support d'opérations frauduleuses comme l'hameçonnage.

•Troisièmement, naguère limité au courrier électronique, le spam gagne maintenant de nouvelles technologies de communication - notamment les appareils mobiles comme les assistants numériques personnels (ANP) et les téléphones intelligents, qui sont de plus en plus utilisés pour accéder aux courriels. En outre, le spam a envahi les services de messagerie instantanée, les blogs et menace le bon

[2] Un PC zombie est un ordinateur mal protégé qui a été infecté par un cheval de Troie et, généralement, une backdoor (porte dérobée).
[3] Un botnet (ou réseau de machines zombies)

fonctionnement des applications de la téléphonie Internet (voir la Figure 1). [1]

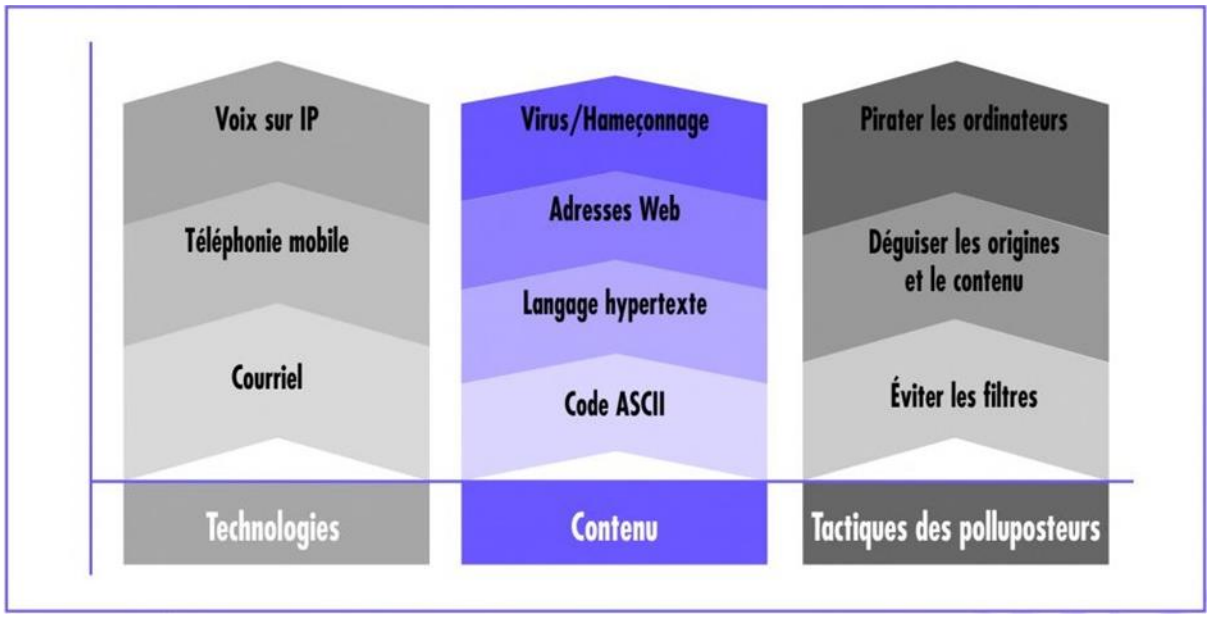


Figure 1 : Evolution du spam

5.Catégories de spam

La plupart des spams ont une vocation commerciale et on peut les répartir en plusieurs catégories:

- les spams commerciaux
- les chaînes de lettres, les avertissements sur les virus, les canulars informatiques
- les emails envoyés par des virus
- les emails d'hameçonnage (ou phishing).

La provenance des adresses email auxquelles des spams sont envoyés est variée, mais quatre grandes catégories se distinguent:

- **Adresses email devinées:** Une grande partie des adresses email est devinée, car il existe des adresses email utilisées dans quasiment tous les domaines, ex : postmaster@votredomainedestrato.fr ou info@nvotredomainedestrato.fr .
- **Récolteur d'adresses (ou moissonneur):** Il s'agit de petits programmes qui visitent des sites web à la recherche d'adresses email, p. ex., dans les livres d'or ou la section Contact des sites web.

- **Jeux concours:** Les jeux concours (dans les centres commerciaux, dans la rue, dans la presse ou sur Internet) sont idéals pour récolter des adresses email. Gardez ce fait à l'esprit et lisez bien les conditions en petits caractères sur les bulletins de participation aux concours.
- **Vendeur d'adresses**: Les données d'adresses, notamment les adresses email, peuvent aussi être acquises auprès de vendeurs d'adresses. Les données d'adresses sont légalement achetées par des entreprises qui les revendent à d'autres sociétés qui les utilisent à des fins publicitaires. [3]

6.Historique de spam

6.1.Premier envoi massif

Premier envoi en masse par la société DEC d'un message à caractère publicitaire aux différents utilisateurs d'ARPANET Le spam original, moins plus de neuf pages, les lignes sont comme suit :

```
Mail-from: DEC-MARLBORO rcvd at 3-May-78 0955-PDT Date: 1 May 1978 1233-EDT
From: THUERK at DEC-MARLBORO Subject: ADRIAN@SRI-KL
DIGITAL WILL BE GIVING A PRODUCT PRESENTATION OF THE NEWEST MEMBERS OF THE
DECSYSTEM-20 FAMILY; THE DECSYSTEM-2020, 2020T, 2060, AND 2060T. THE
DECSYSTEM-20 FAMILY OF COMPUTERS HAS EVOLVED FROM THE TENEX OPERATING SYSTEM
AND THE DECSYSTEM-10 <PDP-10> COMPUTER ARCHITECTURE. BOTH THE DECSYSTEM-2060T
AND 2020T OFFER FULL ARPANET SUPPORT UNDER THE TOPS-20 OPERATING SYSTEM. THE
DECSYSTEM-2060 IS AN UPWARD EXTENSION OF THE CURRENT DECSYSTEM 2040 AND 2050
FAMILY. THE DECSYSTEM-2020 IS A NEW LOW END MEMBER OF THE DECSYSTEM- 20 FAMILY
AND FULLY SOFTWARE COMPATIBLE WITH ALL OF THE OTHER DECSYSTEM-20 MODELS. WE
INVITE YOU TO COME SEE THE 2020 AND HEAR ABOUT THE DECSYSTEM-20 FAMILY AT THE
TWO PRODUCT PRESENTATIONS WE WILL BE GIVING IN CALIFORNIA THIS MONTH.
THE LOCATIONS WILL BE:
TUESDAY, MAY 9, 1978 - 2 PM HYATT HOUSE (NEAR THE L.A. AIRPORT)LOS ANGELES, CA
THURSDAY, MAY 11, 1978 - 2 PM DUNFEY'S ROYAL COACH SAN MATEO, CA(4 MILES SOUTH
OF S.F. AIRPORT AT BAYSHORE, RT 101 AND RT 92) A 2020 WILL BE THERE FOR YOU TO
VIEW. ALSO TERMINALS ON-LINE TO OTHER DECSYSTEM-20 SYSTEMS THROUGH THE
ARPANET. IF YOU ARE UNABLE TO ATTEND, PLEASE FEEL FREE TO CONTACT THE NEAREST
DEC OFFICE FOR MORE INFORMATION ABOUT THE k EXCITING DECSYSTEM-20 FAMILY
```

Figure 2 : Premier envoi massif.

L'expéditeur de ce message était un individu par le nom de Gary Thuerk, qui a travaillé en DEC le département marketing Les réactions au premier spam ont été tout à fait mélangées. Étonnamment, il y avait un peu d'un débat de la justesse du message Utilisations abusives à fins non commerciales, Quelques cas isolés d'utilisations inadéquates de systèmes de messagerie ont été constatées dans les années 80 et jusqu'au début des années 90. Ainsi, une annonce pour la vente d'un service de table est postée en 1985 sur un groupe de discussion usenet[4].

Plus tard en 1993, Richard Depew travaille sur le projet ARMM (Automated Retroactive Minimal Moderation), un système censé protéger les groupes de discussion usenet d'utilisations abusives. Malheureusement, dans le cadre d'un essai d'une version buggée d'ARMM, R. Depew envoie 200 messages sur le groupe news.admin.policy. Face aux récriminations, il s'excuse et utilise le mot «spam » pour désigner ses messages. [2]

6.2.Utilisations abusives à fins commerciales

En mars 1994 Il s'agit d'un cas spectaculaire de spaming lancé des butes strictement commerciales fut celui du cabinet Canter & Siegel. Le message aurait été posté sur près de 6000 groupes de discussion pour un total de 12Mo, ce qui représentait à l'époque environ 10% du trafic quotidien sur usenet[5] En retour, les vives critiques des internautes ont rapidement submergé les installations du cabinet (téléphones, faxes, emails). [2]

5.3.Evolution du spam

Dans les années 1980, beaucoup de lettres de chaîne de courrier électronique ont commencé à apparaître et ont été rapidement détruites. Le tout premier courrier électronique a été enregistrée en février 1982.

Au début des années 1990, la commercialisation d'Internet est finalement arrivée dans la force complète et avec cela est venu tas du spam.

Autour de 1994, les spams plus notables ont commencé à se réaliser. Ce type de spam a été prêté l'attention particulière parce qu'il était le premier à être manifestement abusif de courrier et des systèmes de nouvelles, utilisant le logiciel automatisé pour expédier par la poste (courrier électronique) aux listes. Le premier spam répandu comparable avec des spams d'aujourd'hui a été mentionné comme le spam "de Jésus". Le spam de Jésus a été posté en janvier 1994 à chaque groupe sur Usenet.

[4] Usenet est un système en réseau de forums, inventé en 1979

Probablement le spam le plus notable dans l'histoire, ou au moins le plus parlé, était le Spam de Canter & Siegel. une équipe d'un mari et de femme, des avocats Laurence Canter & Siegel, décidé louer un programmeur qui pourrait écrire le logiciel pour poster une publicité à chaque groupe de discussion déjà existant. Ceci a donné naissance au premier logiciel de diffuseur en vrac connu. Plusieurs permutations différentes du spam ont été envoyées au cours d'une période courte de temps.

Le canter et Siegel n'étaient pas les seuls faisant sortir le spam pendant cette période. Michael Wolff and Company Inc. a décidé de commencer à spammer pour promouvoir certains des livres de Wolff, commençant avec un Chat Net appelé. Wolff a fait sortir environ 150 annonces différentes pour le livre en décembre 1994. Ceux-ci ont été suivis par des publicités pour d'autres livres dans la série, et le spam a commencé à cultivé exponentiellement.

Venez avril 1995, Jeff Slaton, qui s'est appelé le Roi de Spam, a commencé à reprendre l'industrie en inondant des listes de diffusion avec des annonces pour tout de petites entreprises aux annonces politiques.
En août 1995, la toute première liste connue d'adresses électroniques publiques à vendre : 2 millions d'adresses totales.
Pendant les trois ans suivants, on a battu avec le spam dans la force pleine et a été gardé à la baie, au moins en termes de volume.

Entre 1998 et 1999, on battrait le spam cède d'un facteur de 10 à un facteur d'environ 3 ou 4, mais vers la fin de 2000, il commencerait à ramper la sauvegarde de nouveau. Ceci mènerait finalement à une pointe massive dans le volume qui a seulement grandi depuis 2001.

Depuis 2001, le spam a grandi exponentiellement. Vers la fin de 2002, le spam était devenu dans le volume par un facteur de presque 60 comparé à son volume juste six ans antérieurs. Le spam est maintenant tout à fait et complètement hors du contrôle. Au cours des dernières années, des filtres de spam plus complexes ont été conçus et mis en œuvre, utilisant tout de jeux de règle heuristiques de base aux filtres statistiques Les utilisateurs trouvent de plus en plus de spam dans leurs boîtes de réception chaque jour. Beaucoup de chercheurs croient maintenant que le spam est responsable de n'importe où de 35 pour cent à 65 pour cent de tout le trafic de courrier électronique sur Internet aujourd'hui, avec un taux de croissance annuel énorme de 15 pour cent à 20 pour cent. [2]

Conclusion

Nous avons fait un aperçu dans ce chapitre sur le spam, son origine, sa définition, son évolution et ses catégories.

Ce domaine et vaste parce que Le spam représente aujourd'hui, selon les sources, entre 75 et 90 % du trafic e-mail mondial (plus de 100 milliards de pourriels sont envoyés par jour). Et le fléau a aujourd'hui atteint les forums, les blogs, la messagerie instantanée, le fax et les SMS.

Chapitre 02 :

Etat de l'art sur la détection des spams

CHAPITRE II. Etat de l'art sur la détection des spams

Introduction

Dans ce chapitre on va essayer de présenter un état de l'art sur les méthodes inspiré pour la détection des spams, nous allons citer des recherches qui vont nous montrer l'avancement et l'efficacité de détections existants.

1. Utilisation d'un classifieur de spam hybride NB-SVM

En 2018, Jawale Diksha .S et.al proposent l'utilisation d'un classifieur de spam hybride NB-SVM qui utilise les avantages de Naïve Bayes (NB) et Support Vector Machine (SVM), NB est un algorithme de classification rapide et SVM a une grande performance en raison de leur taux de rappel et de précision élevé.

Les données d'apprentissage sont d'abord traitées par l'algorithme NB dans lequel il calcule la probabilité pour chaque mot et message et compare avec un seuil qui classifie Les données. Les données traitées par NB vont à SVM pour améliorer la précision.

L'architecture de ce classifieur est comme suite :

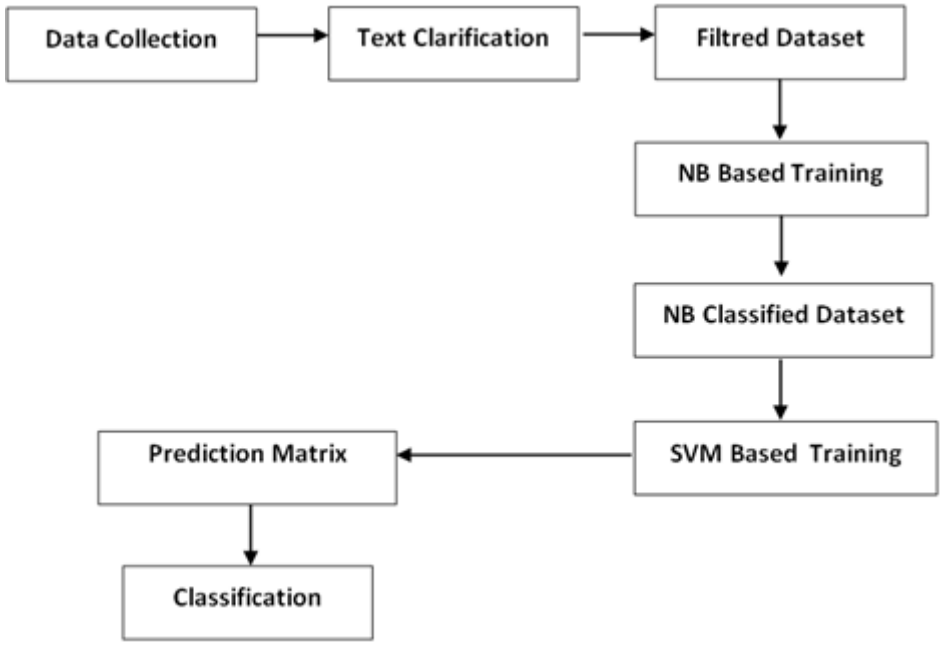


Figure 3 : Architecture NB-SVM

Avec l'utilisation de NB, ils obtiennent une précision de 96,65% dans la phase d'entraînement et 95,78% dans la phase de test. Avec SVM, ils obtiennent une précision de 99,43% dans la phase d'entraînement et 97,13% dans la phase de test. La combinaison de ces deux algorithmes NB-SVM, donne une précision de 99,44% dans la phase d'entraînement et 97,57% dans la phase de test. Ce qui montre que les résultats étaient meilleurs que ceux des deux classifieurs utilisés séparément. [1]

2.Détection de faux avis à l'aide de l'exploration de données

Md Forhad Hossain Août 2019

Dans leurs recherches, ils ont adopté différentes approches pour la détection des avis de spam. ils ont commencé avec la méthode supervisée, puis essayé avec la méthode semi-supervisée et enfin, ils ont utilisé une méthode entièrement non supervisée pour la détection des critiques de spam.

Tout d'abord, ils ont appliqué différents algorithmes d'exploration de données supervisés tels que Support Vector Machine, Naive Bayes et Multi-layer Perceptron. Ils ont découvert que Support Vector Machine donne de meilleurs résultats que Naive Bayes et Multi-layer Perceptron dans la détection de faux avis. Ils ont simulé les travaux d'Ott et al. et obtenu des résultats similaires. Cela nous a donné une précision de près de 90% dans la détection des critiques de spam avec SVM. De plus, Ils ont utilisé l'algorithme Naive Bayes qui offrait près de 87% de précision et le Perceptron multicouche offraient près de 88% de précision dans la détection des avis de spam. Ils ont également recherché une relation entre les parties du discours (POS) et les critiques véridiques / trompeuses. Mais malheureusement, ils n'ont trouvé aucune relation distincte entre les points de vente et les avis véridiques / trompeurs.

Ensuite, Ils ont travaillé avec notre nouvel algorithme semi-supervisé. Ils ont appelé notre approche de détection de spam basée sur la modélisation des sujets. Ils ont appliqué une combinaison de modélisation de sujet et de SVM pour détecter les critiques de spam. Nous avons utilisé des mots de modélisation de sujet comme fonctionnalités pour SVM. votre approche proposée offrait des performances similaires à celles d'Ott et al.malgré l'utilisation des seuls mots de sujet comme

caractéristiques pour SVM alors qu'Ott et al. a utilisé tous les mots dans une critique. votre approche a réduit la dimensionnalité pour SVM et a donné une précision de près de 89% dans la détection des avis de spam. [4]

3.Machine learning pour la détection des courriers indésirables examen, approches et problèmes de recherche ouverts 2019

Dans cette étude, ils ont passé en revue les approches d'apprentissage automatique et leur application au domaine de détection des spams. Un examen de l'état de l'art des algorithmes a été appliqué pour la classification des messages comme spam ou ham est fourni. Les tentatives faites par différents chercheurs pour résoudre le problème du spam grâce à l'utilisation de classificateurs d'apprentissage automatique ont été discutées. L'évolution des messages de spam au fil des ans pour échapper aux filtres a été examinée. L'architecture de base du filtre anti-spam et les processus impliqués dans la détection des courriers indésirables ont été examinés. Le document a étudié certains des ensembles de données et des mesures de performance accessibles au public qui peuvent être utilisés pour mesurer l'efficacité de tout filtre anti-spam. Les défis des algorithmes d'apprentissage automatique pour gérer efficacement la menace du spam ont été soulignés et des études comparatives des techniques d'apprentissage automatique disponibles dans la littérature ont été effectuées. Ils ont également révélé des problèmes de recherche ouverts associés aux filtres anti-spam. En général, le chiffre et le volume de la littérature qu’ils ont examinés montrent que des progrès significatifs ont été accomplis et seront encore réalisés dans ce domaine. Après avoir discuté des problèmes ouverts dans la détection des spams, des recherches supplémentaires pour améliorer l'efficacité des filtres anti-spam doivent être menées. Cela permettra au développement de filtres anti-spam de continuer à être un domaine de recherche actif pour les universitaires et les professionnels de l'industrie qui recherchent des techniques d'apprentissage automatique pour une detection efficace des anti-spam.

[5]

4.Un modèle de détection de spam intelligent basé sur Système immunitaire artificiel 9 juin 2019

Le spam est un problème sérieux qui n'est pas seulement ennuyeux pour les utilisateurs finaux, mais également dommageable financièrement et un risque de sécurité. Les algorithmes et processus d'apprentissage automatique se sont avérés assez efficaces.

Les travaux présentés dans cet article, basés sur des systèmes de détection d'anomalies et des principes d'apprentissage automatique,
démontre que l'inclusion de plus de bases de données augmente considérablement le taux de détection correct, de 93,14% basé sur un ensemble de données à près de 98,57% après avoir inclus le dernier ensemble de données (Enron[6]). Le la fusion du détection basé sur IP avec un algorithme de sélection négative dans une approche supervisée est un approche novatrice.

Ils ont également comparé notre système proposé à d'autres systèmes utilisant NSA et constaté que notre Le système proposé surpasse les autres applications de la NSA en termes de détection réelle de spam et de ham. [6]

5.Un spam à vie modèle de classification

Rami Mustafa A. Mohammad 20 janvier 2020

Dans cet article, un modèle de classification des spams à vie a été créé. Un tel modèle est appelé «Classification à vie basée sur un ensemble utilisant le partitionnement réglable des ensembles de données». La propriété permanente du modèle a été obtenue en s'assurant que le modèle est capable de gérer la dérive de concept et les dilemmes catastrophiques d'oubli qui sont généralement présumés

[6] Le corpus Enron est une base de données de plus de 600 000 emails générés par 158 employés d'Enron Corporation au cours des années qui ont précédé l'effondrement de l'entreprise en décembre 2001. Le corpus a été généré à partir des serveurs de messagerie d'Enron par la Federal Energy Regulatory Commission (FERC) au cours de sa enquête ultérieure.

être les principaux défis lors de la création de tout système DM et ML. L'ELCADP utilise les informations obtenues de l'EDDM qui est une méthode de détection de concept bien connue. Les informations obtenues à partir de l'EDDM fournissent des informations sur le niveau de dérive du concept. Niveau de dérive du concept, l'ELCADP se prononce alors sur le nombre de partitions nécessaires à la création d'un nouveau modèle de classification. Pour les besoins de l'évaluation, le jeu de données bien connu «EnronSpam» est utilisé. L'évaluation empirique a montré que l'ELCADP[7] surpasse toutes les méthodes de minage fluvial utilisées à des fins de comparaison. Quatre paramètres d'évaluation ont été utilisés pour évaluer les performances de l'ELCADP: l'exactitude, la précision, le rappel et le score F1. L'ELCADP a montré des résultats solides dans toutes ces mesures d'évaluation et qui confirment que l'ELCADP a été en mesure de créer un modèle de classification de spam à vie. Dans l'ensemble, cette étude de recherche a montré que les systèmes de classification traditionnels des spams hors ligne pourraient ne pas être le bon choix pour créer des systèmes de classification à vie. Cependant, il convient de mentionner ici que la performance ELCADP n'a pas été examinée dans le cas où la dérive de concept qui pourrait se produire est une dérive de concept virtuel où une nouvelle valeur de classe pourrait apparaître alors que les caractéristiques d'entrée sont toujours inchangées. Ceci est en fait laissé comme un travail futur et la classification des sites de phishing est un domaine possible pour examiner la capacité de l'ELCADP à gérer la dérive du concept virtuel qui caractérise le problème de classification des sites de phishing. [7]

6.Détection efficace du spam d'opinion

Une étude sur Revoir les métadonnées par rapport au contenu 23 avril 2020

Dans cet article, ils ont étudié l'impact de différentes catégories de caractéristiques sur l'opinion détection de spam. En substance, ils ont cherché à identifier ces caractéristiques qui jouent le rôle le plus dominant en étant les signatures de spam qui définissent. Pour la même chose, ils ont examiné l'efficacité des fonctionnalités comportementales et textuelles conçues à partir de deux ensembles de données de référence YelpZip et YelpNYC.

[7] Ensuring Lifelong spam Classification model using Adjustable Dataset Partitioning.

Pour comprendre leur polyvalence, cet examen a été mené dans trois contextes de spam différents, à savoir les paramètres d'examen, de réviseur et de produit, chaque paramètre étant essentiel en soi. Dans le contexte du spam, chaque paramètre est spécifique à la cible, où les transactions centrées sur la révision traitent du spam de révision, les transactions centrées sur les réviseurs avec les spammeurs de révision et les produits centrés sur les produits ciblés sur le spam. Nous avons effectué l'ingénierie des fonctionnalités de manière indépendante et proposé de nouvelles fonctionnalités comportementales et textuelles sous trois paramètres prédéfinis.

Pour examiner l'efficacité de ces fonctionnalités dans la détection du spam d'opinion, nous avons formé quatre classificateurs supervisés, à savoir SVM, LR, MLP et NB sur les deux jeux de données sous les paramètres mentionnés ci-dessus.

Examiner l'efficacité de l'ensemble proposé de fonctionnalités, ils ont ensuite comparées à certains travaux connexes bien connus antérieurs dans les trois paramètres.

Leurs résultats indiquent que les caractéristiques comportementales surpassent le texte dans les trois paramètres sur les deux ensembles de données. [8]

7.Approche de détection de spam pour un message mobile sécurisé Communication à l'aide d'algorithmes d'apprentissage automatique 6 juin 2020

La détection précise du spam est un gros problème, et de nombreuses méthodes de détection ont été proposées par divers chercheurs. Cependant, ces méthodes n'ont pas la capacité de détecter le spam avec précision et efficacité. Pour résoudre ce problème, nous avons proposé une méthode de détection de spam utilisant le machine learning prédictif des modèles (arbre de dicision , knn , LR).

La méthode est appliquée à des fins de détection de spam.

Les résultats expérimentaux obtenus montrent que la méthode proposée a une grande capacité à détecter le spam. la méthode proposée a atteint une précision de 99% qui est élevé par rapport aux autres méthodes existantes.
ainsi, les résultats suggèrent que la méthode proposée est plus fiable pour une détection précise et ponctuelle du spam, et qu'elle sécurisera les systèmes de communication des messages et des emails. [9]

8.Detecting Web Spam Based on Novel Features From Web Page Source Code Jiayong Liu, 1 Yu Su, 1 Shun Lv, 2 et Cheng Huang

4 décembre 2020

Sur la base des recherches actuelles, cet article propose une nouvelle méthode pour distinguer le spam Web. Ils introduisent un ensemble de nouvelles fonctionnalités sur la page d'accueil qu’ils ont vérifiées manuellement. En attendant, ils utilisent la sélection de fonctionnalités algorithme Smart-BT pour réduire la dimension des entités existantes pré calculées afin que le coût de calcul de la méthode diminue. Ensuite, ils utilisent le modèle RF pour discriminer le spam Web avec une identification efficace. Les résultats de l'expérience ont montré que cette méthode pouvait atteindre un niveau de pointe par rapport à d'autres méthodes.

Le modèle avec des nouvelles fonctionnalités qui sont impressionnantes pour la détection de spam Web est plus supérieur et valide que le modèle avec uniquement des fonctionnalités existantes. Comme cet article ne prend en compte que la page d'accueil, la méthode est générale et extensible car obtenir toutes les pages d'un site Web n'est pas facile dans la plupart des cas. Ils reconnaissent que certains des biais de notre ensemble de données peuvent affecter le résultat. Leur méthode peut ne pas fonctionner correctement à mesure que le spam Web évolue, car la frontière entre le spam et le non-spam est susceptible de s'estomper. De plus, ils n'ont analysé statiquement qu'à partir du code source sans tenir compte des

parties dynamiques telles que le code JavaScript, leur méthode a donc des limites pour le spam Web qui utilise la technologie dynamique. Par exemple, le camouflage et la redirection du spam Web. la méthode proposée se concentre uniquement sur la page d'accueil d'un certain site Web sans confirmer si le site Web renvoie un contenu différent pour les utilisateurs et les moteurs de recherche, de sorte qu'il y a une certaine erreur dans la détection de ce type de spam Web. [10]

Conclusion :

Parmi ces publication chacun de ces chercheurs sa méthode et son domaine applicatif. Dans la littérature il existe de nombreux travaux qui traitent le problème de détection des spams par l'utilisation des méthodes d'apprentissage automatique.

La détection des spams basé sur le contenu textuel des messages peut être considéré comme un exemple de classification de textes qui consiste en l'attribution de documents textuels à un ensemble de classes prédéfinis.

Chapitre 03 :

L'apprentissage automatique

Chapitre III L'apprentissage automatique

Introduction

En une vingtaine d'années, l'apprentissage artificiel est devenu une branche majeure des mathématiques appliquées, à l'intersection des statistiques et de l'intelligence artificielle. Son objectif est de réaliser des modèles qui apprennent par des exemples: il s'appuie sur des données numériques (résultats de mesures ou de simulations). L'objectif des chercheurs en intelligence artificielle vise à programmer des machines capables d'effectuer des tâches qui requièrent de l'intelligence. Ces domaines d'applications sont multiples: fouille de données (FD), bioinformatique, génie des procédés, aide au diagnostic médical, télécommunications, interface cerveau-machines, et bien d'autres.

1.Différents types d'apprentissage

1.1.Apprentissage supervisé

Il relève d'une démarche inductive consistant à construire automatiquement un classifieur qui apprend, à partir des exemples déjà classés (ou étiquetés), les caractéristiques et les propriétés des catégories cibles. Ce type d'apprentissage est dit supervisé par ce que la fonction de classification s'entraine sur les catégories (ou classe) ainsi que sur leurs caractéristiques.

La classification supervisée cherche à prédire l'appartenance de documents à des classes connues à priori. Ainsi, c'est l'ensemble des techniques qui visent à deviner l'appartenance d'un individu à une classe.

1.2.Apprentissage non supervisé

Dans ce type d'apprentissage, il n'existe pas de classes prédéfinies, le but est d'effectuer les meilleurs regroupements possibles, entre les objets dans lesquels, les observations diffèrent très peu, au regard de ses valeurs.

Le plus connu des problèmes non-supervisés est la classification non-supervisée ou clustering. Les classes qu'on appellera clusters, sont formées par regroupement des données qui ont certaines caractéristiques en commun.

Pour construire un regroupement de ces données, nous avons trois choix à faire :

- Choisir une mesure de ressemblance (ou similarité) entre les données
- Choisir le type de structures que nous voulons obtenir : partition, hiérarchie, ..
- Choisir la méthode permettant d'obtenir la structure désirée

1.3.Apprentissage semi supervisé

Dans l'apprentissage semi supervisé, certaines données sont étiquetées et d'autres ne le sont pas. Il réalise les mêmes tâches que celles réalisées en apprentissage supervisé, à la différence qu'il fait usage des données non étiquetées.

2.Les différents méthodes de Apprentissage supervisé

Il existe différents méthodes :

2.1.Les k plus proches voisins

Les k plus proches voisins K-nearest neighbors (K-NN) est une méthode d'apprentissage supervisée qui résonnent avec le principe sous-jacent. Elle diffère des autres méthodes d'apprentissages par l'absence du modèle induit par des exemples. Les données restent telles quelles, elles sont simplement stockées en mémoire. Selon le nombre k choisi, un nouvel exemple sera classé dans la classe majoritaire pour les k voisins sélectionnés.

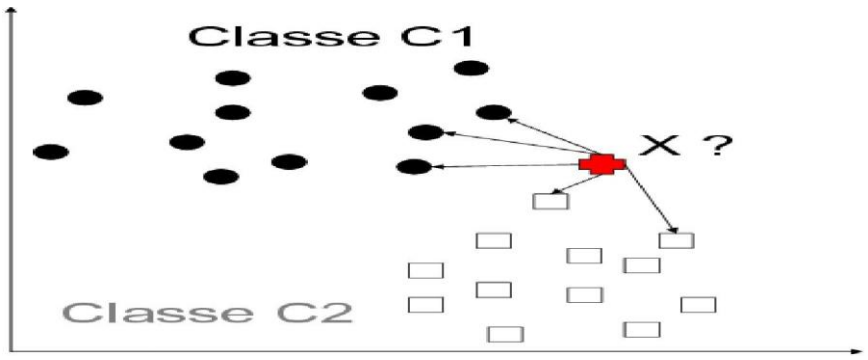


Figure 4 : Les k plus proches voisins.

Avantages

- Absence d'apprentissage: Ce sont les échantillons pris en considération, qui constituent le modèle.
- Clarté des résultats: bien que la méthode ne produise pas de règle explicite, la classe attribuée à un exemple peut être expliquée en exposant les plus proches voisins qui ont imposé cette attribution.
- Données hétérogènes: la méthode peut s'appliquer dès qu'il est possible de définir une distance sur les champs.
- Grand nombre d'attributs: la méthode permet de traiter des problèmes avec un grand nombre d'attributs. Cependant, plus le nombre d'attributs est important, plus le nombre d'exemples doit être grand.

Inconvénients

- Sélection des attributs pertinents: pour que la notion de proximité soit pertinente, il faut que les exemples couvrent bien l'espace et soient suffisamment proches les uns des autres. Si le nombre d'attributs pertinents est faible relativement au nombre total d'attributs, la méthode donnera de mauvais résultats.
- Le temps de classification: si la méthode ne nécessite pas d'apprentissage, tous les calculs doivent être effectués lors de la classification.
- Définir les distances et nombres de voisins: les performances de la méthode dépendent du choix de la distance, du nombre de voisins et du mode de combinaison des réponses des voisins. Le calcul des distances euclidiennes impose que les attributs constituent un espace orthonormé.

2.2.Réseaux de neurones

Les réseaux de neurones forment une classe de classifieurs supervisés et non supervisés. Ils sont inspirés de la structure neurophysiologique des neurones. Un neurone formel est l'unité élémentaire d'un système modélisé d'un réseau de neurone. A la réception de signaux provenant d'autres neurones du réseau, un neurone formel réagit en produisant un signal de sortie qui sera transmis à d'autres neurones du réseau. Le signal reçu à la sortie d'un neurone est une combinaison linéaire des sorties et neurones précédents. Le signal de sortie est une fonction de cette somme pondérée :

$$y_j = f(\sum^{R}_{i=1} w_{ij}. x_i)$$

Avec y_j la sortie du neurone formelle j, x_i les signaux reçus par le neurone j de la part des neurones i, w_{ij} les poids des interconnexions entre les neurones i et j. Selon l'application, la fonction f, appelée fonction d'activation, est le plus souvent une fonction identité, sigmoïde, tangente hyperbolique ou une fonction linéaire par morceaux.

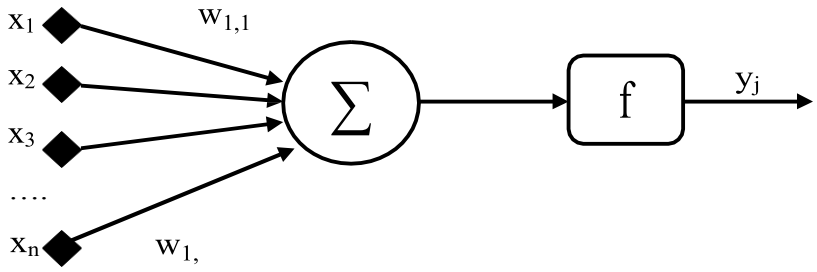


Figure 5 : Neurone formel.

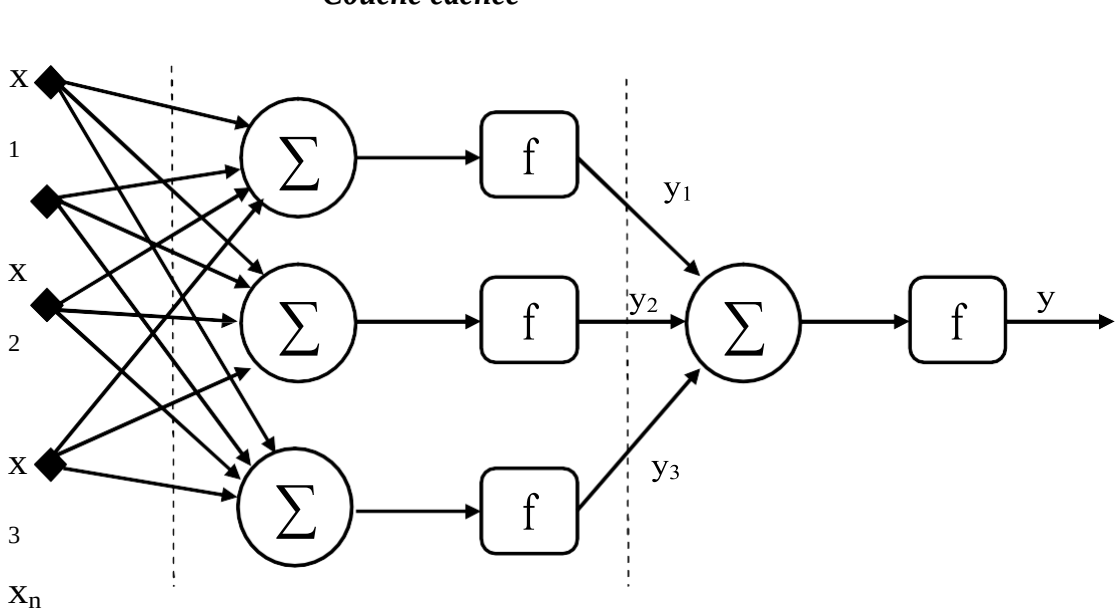


Figure 6 : Perceptron multicouche .

Avantages

- Lisibilité du résultat : le résultat de l'apprentissage est un réseau constitué de cellules organisées selon une architecture, définies par une fonction d'activation et un très grand nombre de poids à valeurs réelles.
- Les données réelles : les réseaux traitent facilement les données réelles "préalablement normalisées" et les algorithmes sont robustes au bruit.
- Classification efficace : le calcul d'une sortie à partir d'un vecteur d'entrée est un calcul très rapide.
- Leur utilisation est diverse. En plus des tâches de classification, les RNN peuvent être utilisés en détection, en modélisation…etc.

Inconvénients

- Détermination de l'architecture du réseau est complexe.
- Paramètres difficiles à interpréter (boite noire).
- Difficulté de paramétrage surtout pour le nombre de neurone dans la couche cachée.
- Temps d'apprentissage:l'échantillon nécessaire à l'apprentissage doit être suffisamment grand et représentatif des sorties attendues.

2.3.Arbres de décision

Les arbres de décision sont les plus populaires des méthodes d'apprentissage. Comme toute méthode d'apprentissage supervisé, les arbres de décision utilisent des exemples. Si l'on doit classer des exemples dans des catégories, il faut construire un arbre de décision par catégorie. Pour déterminer à quelle catégorie appartient un nouvel exemple, on utilise l'arbre de décision de chaque catégorie auquel on soumet le nouvel exemple à classer.

Chaque arbre répond Oui ou Non alors il prend une décision. Concrètement, chaque nœud d'un arbre de décision contient un test conditionnel et les feuilles ont les valeurs Oui ou Non. Chaque test regarde la valeur d'un attribut de chaque exemple. Pour construire l'arbre de décision, il faut trouver quel attribut tester à chaque nœud.

C'est un processus récursif. Pour déterminer quel attribut tester à chaque étape, on utilise un calcul statistique qui détermine dans quelle mesure cet attribut sépare bien les exemples Oui/Non. On crée alors un nœud contenant ce test, et on crée autant de descendants que de valeurs possibles pour ce test.

L'arbre de décision classe trop bien les exemples, mais est mauvais pour généraliser, c'est-à-dire qu'il prédit mal la classification (Oui /Non) de nouvelles instances.

Soit A, B et C trois entiers, pour les ordonnées par un arbre de décision, un ensemble de test doit être réalisé, la figure I.8 illustre cet exemple :

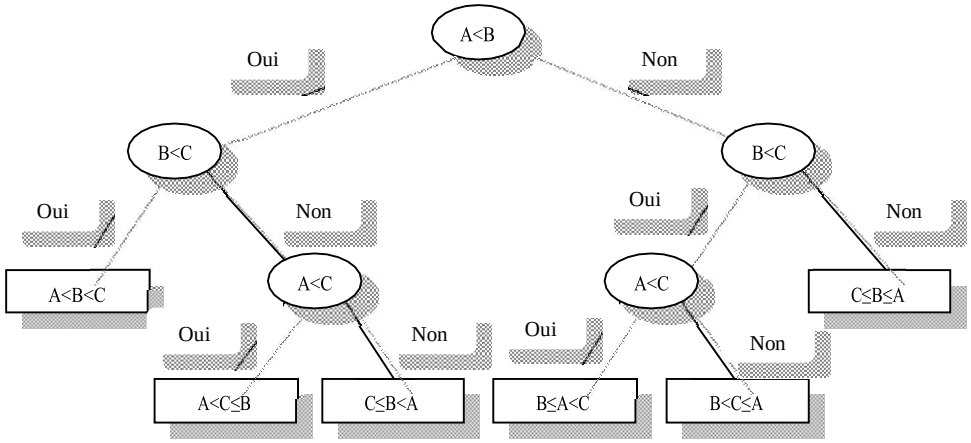


Figure 7 : Ordonnancement de trois entiers par Arbre de décision.

Avantages

- Adaptabilité aux attributs de valeurs manquantes : les algorithmes peuvent traiter les valeurs manquantes (descriptions contenant des champs non renseignés) pour l'apprentissage, mais aussi pour la classification.
- Bonne lisibilité du résultat : un arbre de décision est facile à interpréter et à la représentation graphique d'un ensemble de règles. Si la taille de l'arbre est importante, il est difficile d'appréhender l'arbre dans sa globalité. Cependant, les outils actuels permettent une navigation aisée dans l'arbre (parcourir une branche, développer un nœud, élaguer une branche) et, le plus important, est certainement de pouvoir expliquer comment est classé un exemple par l'arbre, ce qui peut être fait en montrant le chemin de la racine à la feuille pour l'exemple courant.
- Traitement de tout type de données : l'algorithme peut prendre en compte tous les types d'attributs et les valeurs manquantes. Il est robuste au bruit.
- Sélectionne des variables pertinentes : l'arbre contient les attributs utiles pour

la classification. L'algorithme peut donc être utilisé comme prétraitement qui permet de sélectionner l'ensemble des variables pertinentes pour ensuite appliquer une autre méthode.

- Donne une classification efficace : l'attribution d'une classe à un exemple à l'aide d'un arbre de décision est un processus très efficace (parcours d'un chemin dans un arbre).
- Disponibilité des outils : les algorithmes de génération d'arbres de décision sont disponibles dans tous les environnements de fouille de données.
- Méthode extensible et modifiable : la méthode peut être adaptée pour résoudre des tâches d'estimation et de prédiction. Des améliorations des performances des algorithmes de base sont possibles grâce aux techniques qui génèrent un ensemble d'arbres votant pour attribuer la classe.

Inconvénients

- Méthode sensible au nombre de classes : les performances tendent à se dégrader lorsque le nombre de classes devient trop important.
- Manque d'évolutivité dans le temps : l'algorithme n'est pas incrémental, c'est-à-dire, que si les données évoluent avec le temps, il est nécessaire de relancer une phase d'apprentissage sur l'échantillon complet (anciens exemples et nouveaux exemples).

2.4.Support Vector Machines

Les Support Vector Machines constituent une technique d'apprentissage supervisée introduite en fin des années 90. Grâce à son fondement mathématique et à ses performances, cette technique a ouvert un domaine de recherche très actif et un grand éventail d'applications.

Le SVM consiste à chercher le meilleur hyperplan qui sépare linéairement deux classes tout en les repoussant au maximum. Lors de sa phase d'apprentissage, le SVM vise à maximiser la marge entre les deux classes d'apprentissage. Ce qui lui procure un grand pouvoir de généralisation pendant la phase de test. Cette méthode sera détaillée au chapitre suivant.

Avantages

- Les SVM possèdent des fondements mathématiques solides.
- Les exemples de test sont comparés juste avec les supports vecteur et non pas avec tout les exemples d'apprentissage.
- Décision rapide. La classification d'un nouvel exemple consiste à voir le signe de la fonction de décision f(x).

Inconvénients

- Classification binaire d'où la nécessité d'utiliser l'approche un-contre-un pour construire un classifieur multiclasse.
- Grande quantité d'exemples en entrées implique un calcul matriciel important.
- Temps de calcul élevé lors d'une régularisation des paramètres de la fonction noyau.

Conclusion

Dans ce chapitre on a présenté quelques méthodes d’apprentissage automatique avec ces avantages et ces Inconvénients, on va choisir apprentissage automatique par SVM.

Chapitre 04 :

Support Vector Machines

CHAPITRE IV. Support Vector Machines

Introduction

Les "Support Vector Machines", ou Séparateurs à Vaste Marge (SVM) sont un ensemble de techniques d'apprentissage supervisée destinés à résoudre les problèmes de discrimination et de régression. Initiée par Vladimir Vapnik comme méthode de classification binaire qui cherche le meilleur hyperplan qui sépare linéairement les exemples positifs des exemples négatifs en garantissant que la marge entre le plus proche des positifs et des négatifs soit maximal. Ils sont ensuite développés par l'introduction de fonction dites noyau (kernel) par Boser afin de régler les problèmes de la non linéarité, quant à Cortes & al pour traiter les cas de données non linéairement séparables proposent une version régularisée des SVM qui tolère les erreurs d'apprentissage tout en les pénalisant.

1.Notions de bases

1.1.Hyperplan

Quand on est dans un espace de représentation euclidien, on peut librement faire des hypothèses sur la géométrie des classes ou sur celles de leurs surfaces séparatrices ; ceci permet de mettre au point des techniques d'apprentissage non statistiquement fondées a priori, mais peut être plus faciles à appliquer. La plus simple d'entre elles est de supposer que deux classes peuvent être séparées par une certaine surface. Les paramètres qui régissent son équation sont alors les variables à apprendre. Le nombre de paramètres à calculer est minimal si l'on suppose cette surface linéaire. Dans R^p; une surface linéaire est un hyperplan H; défini par l'équation : $w^T x + b = 0$; Si deux classes C1 et C2 sont séparables par H, alors tous les points de la première classe sont par exemple tels que : $x \in C_1 \Rightarrow w^T x + b \geq 0$ et de la seconde vérifient alors : $x \in C_2 \Rightarrow w^T x + b < 0$.

On parle d'hyperplan optimal lorsque celui-ci sépare les deux classes en garantissant un maximum d'espace entre les vecteurs de support.

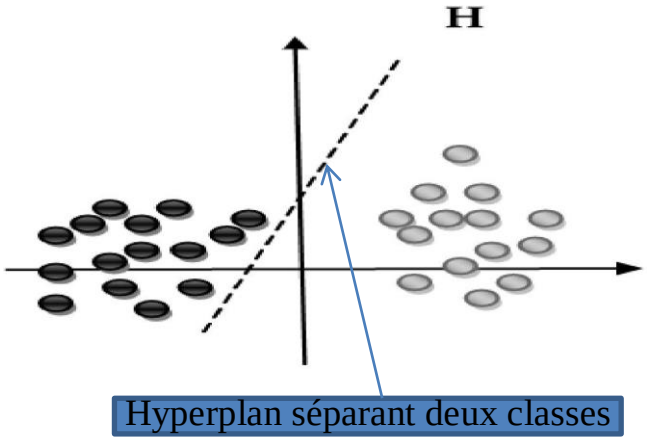


Figure 8 : Support vectors.

Ce sont les points les plus proches de l'hyperplan optimal et qui déterminent la marge.

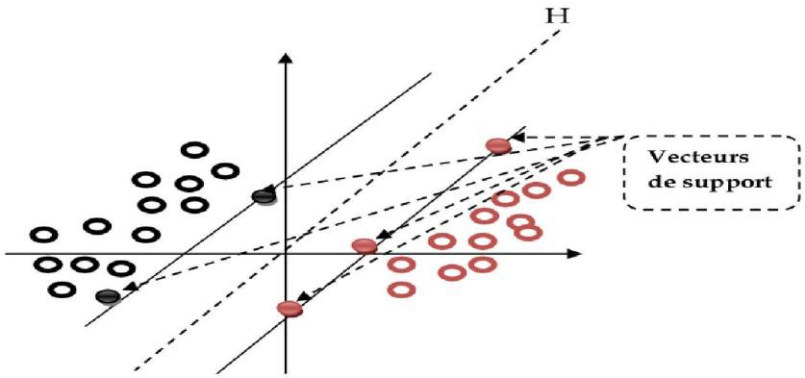


Figure 9 : Support vectors. avec les vecteurs de support.

1.2.Marge

La marge est la distance euclidienne entre deux vecteurs de support de deux classes différentes.

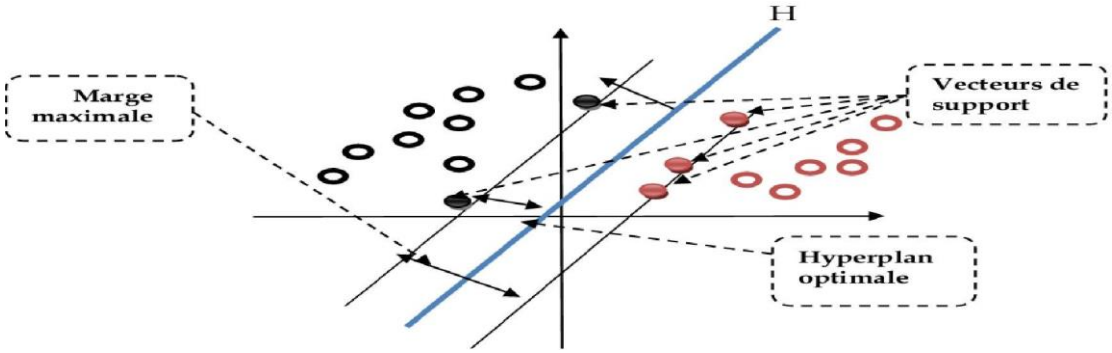


Figure 10 : Représentation de la marge

2.Théorie d'apprentissage de Vapnik-Chervonenkis

Afin de résoudre le problème d'apprentissage qui est basé sur la minimisation du risque réel, et vu l'insuffisance de la MRE, Vladimir Vapnik et Chervonenkis ont développé cette théorie[8] où ils ont montré que

- La condition nécessaire et suffisante pour la consistance de principe MRE est que h soit finie.
- Si F possède une dimension VC finie h, que m>h, avec une probabilité d'erreur au moins égale à 1-5, l'inégalité suivante sera vérifiée :

$$Rréel \leq Remp + \sqrt{\frac{h\left[ln\left(\frac{2m}{h}\right)+1\right]-ln\left(\frac{\eta}{4}\right)}{m}} \quad \textbf{I1}$$

Le membre droit de l'inégalité appelé le risque garanti est composé de deux termes :

[8] Une théorie mathématique et informatique développée dans les années 1960-1990 par Vladimir Vapnik et Alexey Chervonenkis. C'est une forme de théorie de l'apprentissage automatique, qui tente d'expliquer l'apprentissage d'un point de vue statistique.

Le risque empirique et une quantité qui dépond du rapport $\frac{m}{h}$ Appelèe intervalle de confiance puisqu'il représente la différence entre le risque empirique $\boldsymbol{R}_{\text{emp}}$ et le risque Rréel.

L'insuffisance de la minimisation du risque empirique **MRE** a incité **VC** de proposer un nouveau principe d'induction « Minimisation du risque structurel » qui a pour but la minimisation du risque réel tout en minimisant conjointement le Remp et l'intervalle de confiance ou la classe de fonction ;

D'une part, en limitant fortement la taille de l'ensemble **H**, on tend à expliquer la relation entre les objets et leurs classes, on parle d'une sur généralisation. Dans ce cas le risque empirique sera élevé mais le modèle ne collera pas aux exemples de trop près.

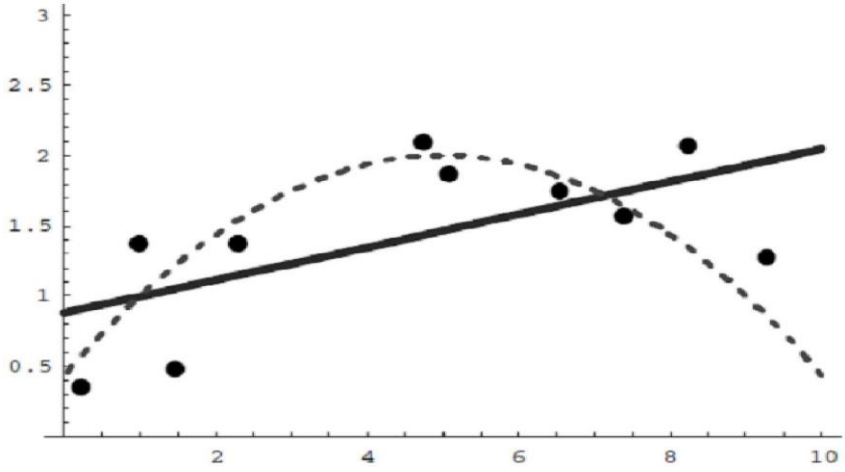


Figure 11 : Sous apprentissage

D'autre part, si on admet un grand nombre de fonctions, la relation sera modélisée de manière complexe et le bruit associé aux mesures risque également d'être appris. On parle souvent d'apprentissage par cœur parce que le classifieur aura un risque empirique très faible mais ses performances sur d'autres jeux de données seront mauvaises.

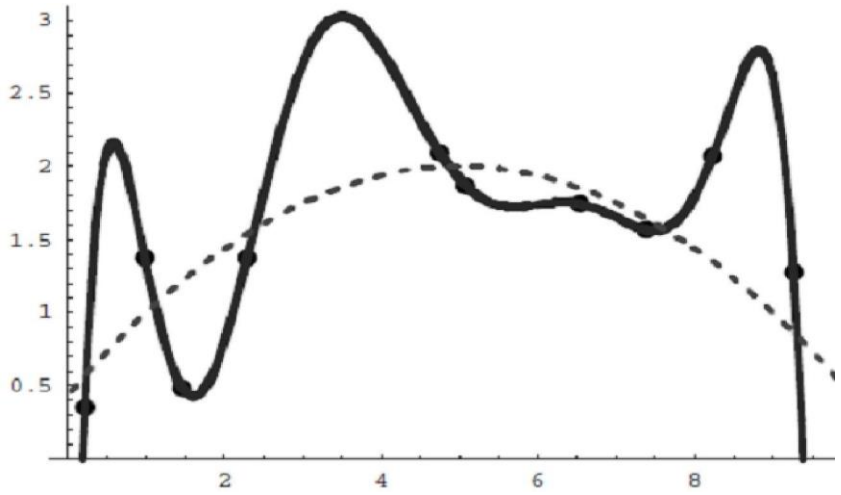


Figure 12 : Apprentissage par cœur

La restriction des fonctions implémentables nous confronte à un dilemme que les statisticiens appellent biais-variance.

La méthode de minimisation de risque structurel **MRS** cherche alors un compromis entre la taille de classe de fonction qui réalise l'approximation et l'approximation sur l'échantillon. L'interaction de la courbe de confiance (variance) et du risque empirique (biais) qui nous donnera la minimisation du risque réel.

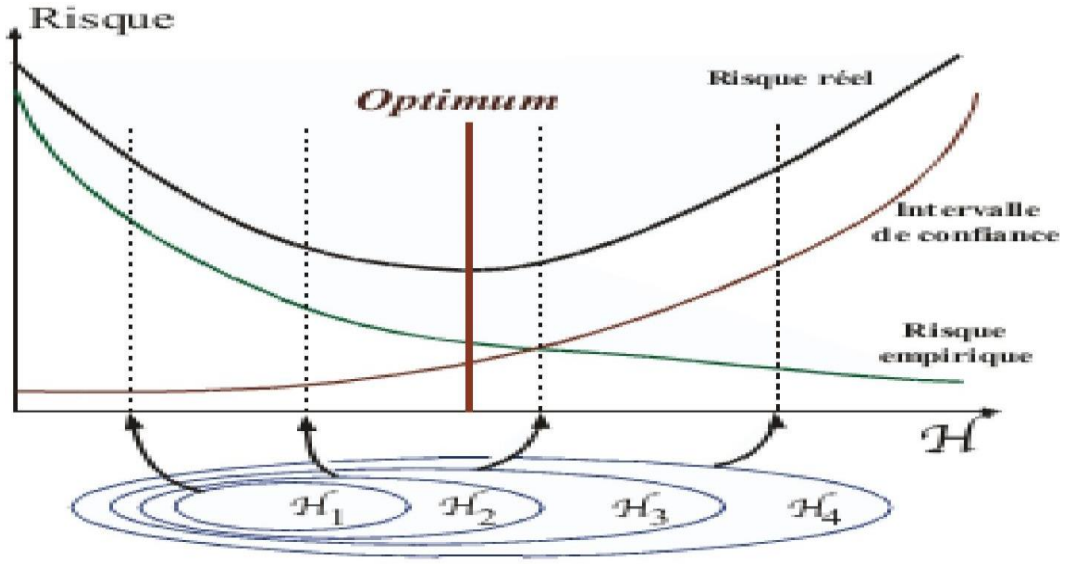


Figure 13 : Comportement du risque empirique.
L'intervalle de confiance et le risque garanti en fonction de la VC dimension.

3.Principe d'unSVM

Cette technique est une méthode de classification à deux classes. Son objectif est non seulement de séparer les exemples positifs des exemples négatifs mais aussi de repousser au maximum les uns des autres. La méthode cherche alors l'hyperplan qui sépare les deux classes d'exemples, en garantissant que la marge entre les positifs et les négatifs les plus proches de l'hyperplan soit maximale. Cela garantit une meilleure généralisation du modèle car de nouveaux exemples pourraient ne pas être trop similaires à ceux utilisés pour l'apprentissage. L'intérêt de cette méthode est la sélection de vecteurs supports qui représentent les vecteurs discriminant grâce auxquels est déterminé l'hyperplan. Seuls ses vecteurs interviennent dans la solution, ce qui rend le problème moins complexe.

3.1.Principes fondamentaux

a. Maximisation de la marge

Intuitivement, le fait d'avoir une marge plus large procure plus de sécurité lorsqu'on classe un nouvel exemple. De plus, si l'on trouve le classificateur qui se comporte le mieux vis-à-vis des données d'apprentissage, il est clair qu'il sera aussi celui qui permettra au mieux de classer les nouveaux exemples.

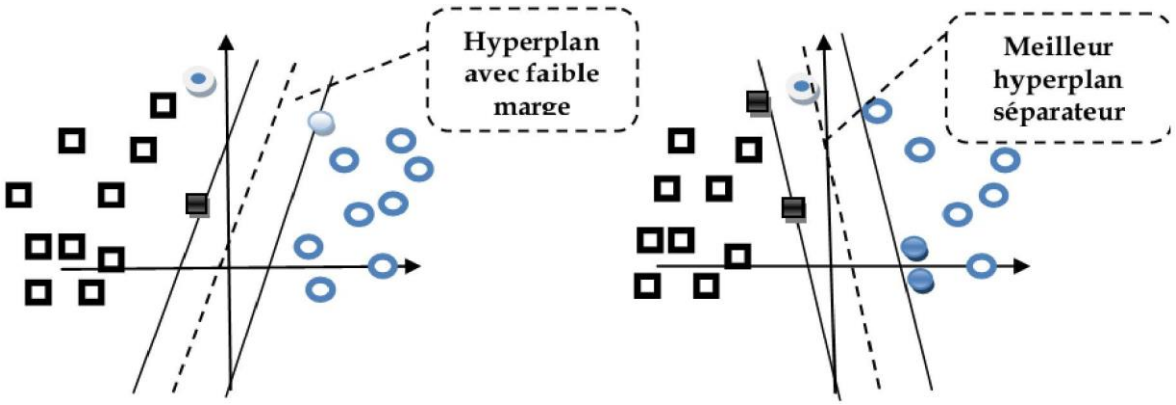


Figure 14 : Meilleur hyperplan séparateur.

b.Cas linéairementséparable

Les cas linéairement séparables sont les plus simples des SVM car ils permettent de trouver facilement le classificateur linéaire.

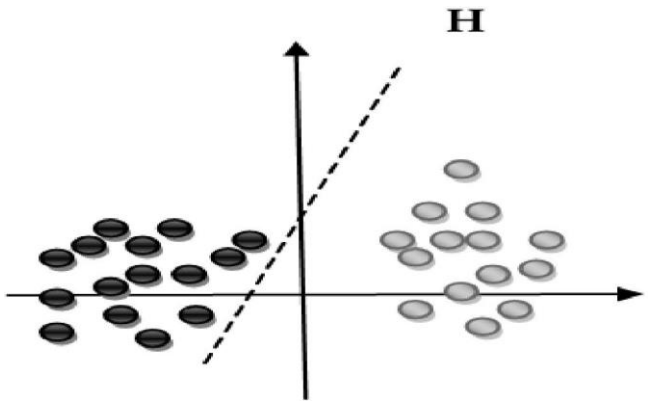


Figure 15 : Exemple d'un cas linéairement séparable

Dans la plupart des problèmes réels il n'y a pas de séparation linéaire possible entre les données, le classificateur de marge maximale ne peut pas être utilisé car il fonctionne seulement si les classes de données d'apprentissage sont linéairement séparables.

c.Cas non linéairement séparable

Pour surmonter les inconvénients des cas non linéairement séparable, l'idée introduite par V.Vapnick et qui fait d'ailleurs le point fort des SVMs est de projeter l'espace d'entrée sur un espace de plus grande dimension où les données deviennent linéairement séparables. Ce nouvel espace est appelé « espace de redescription ». Intuitivement, plus la dimension de l'espace de redescription est grande, plus la probabilité de pouvoir trouver un hyperplan séparateur entre les exemples est élevée.

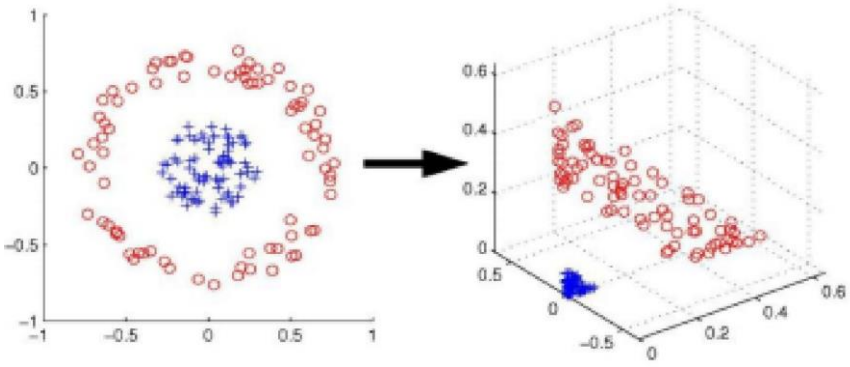


Figure 16 : Exemple de projection dans un espace de redescription

La figure 9 montre un exemple de transformation d'un problème de séparation non linéaire dans l'espace de représentation (2 dimensions) en un problème de séparation linéaire dans un espace de redescription de plus grande dimension (3 dimensions). Cette transformation non linéaire est réalisée via une fonction noyau.

En pratique, quelques familles de fonctions noyau paramétrables sont connues et il revient à l'utilisateur du SVM d'effectuer des tests pour déterminer celle qui convient le mieux pour son application. On peut citer les exemples de noyaux suivants : polynomial, gaussien, sigmoïde et Laplacien.

3.2.Fondement mathématique :

3.2.1.Cas linéairementséparable :

Un classifieur est dit linéaire lorsqu'il est possible d'exprimer sa fonction de décision par une fonction linéaire en x. On peut exprimer une telle fonction par:

$$h(x_i) =< w , x_i > +b = \sum_{j=1}^{p} w_j . x_i^j + b \qquad \textbf{I2}$$

Où w est le vecteur de poids et b le biais, alors que x est la variable. X est l'espace d'entrée qui correspond à R^p, où p est le nombre d'attributs des

vecteurs d'entrée. P est également la dimension de l'espace d'entée. Notons que l'opérateur < > désigne le produit scalaire usuel.

Pour décider à quelle classe un exemple appartient, il suffit de prendre le signe de la fonction de décision :

$$\begin{cases} (w.xi) + b \geq 1 \text{ si } yi = 1 \\ (w.\ xi) + b \leq -1 \text{ si } yi = -1 \end{cases}$$

Ce qui est équivalent à :

$$yi((w.\ x) + b) \geq 1 \qquad \textit{avec i=1…n} \qquad \textbf{I3}$$

Trouver l'hyperplan optimal revient à maximiser la marge M et donc à maximiser la somme des distances euclidienne (d) des deux classes par rapport à l'hyperplan. Ainsi, la marge est donnée par l'expression suivante :

$x_b=x_a-d\frac{w}{\|w\|}$ avec x_b ,x_a ,w sont des vacteurs et $\frac{w}{\|w\|}$ vecteur unitaire b étant un point de l'hyperplan h ,alors il satisfait l'équation

$$(w.x_b) + b = 0 \qquad \textbf{I4}$$

Dans ce cas, pour x_a qui est un point qui n'appartiens pas à l'hyperplan on aura donc :

$$\langle w(x_a - d\frac{\|w\|}{\|w\|})\rangle + b = 0 \qquad \textbf{I5}$$

$$\langle w.x_a\rangle - d\frac{\|w.w\|}{\|w\|} + b = 0 \qquad \textbf{I6}$$

$$\langle w.x_a\rangle - d\|w\| + b = 0 \qquad \textbf{I7}$$

$$d = \frac{(\langle w.x_a\rangle + b}{\|w\|} \qquad \textbf{I8}$$

$$|\langle w.x_a\rangle + b| = 1 \qquad \textbf{I9}$$

$$d = \frac{1}{\|w\|} \qquad \textbf{I10}$$

$$M = 2.d = \frac{2}{\|w\|} \qquad \textbf{I11}$$

Le problème devient :

$$PQ1 \begin{cases} \min \frac{1}{2} * \|w\|^2 + c\sum_{i=0}^{n} \mathcal{E}i \\ SCs \\ Sc \quad y_i[\langle w . x_i \rangle + b] \geq 1 - \mathcal{E}_i \end{cases} \qquad \textbf{I12}$$

Avec i = 1 … n.

a-Marge dure (tous les $\mathcal{E}_{i\,i}$ = 0 et C = 0 dans (PQ1))

Il s'agit d'un problème quadratique convexe sous contraintes linéaires de forme primal dont la fonction objective est à minimiser. Cette fonction objective est le carré de l'inverse de la double marge. L'unique contrainte stipule que les exemples doivent être bien classés et qu'ils ne dépassent pas les hyperplans canoniques.

Dans cette formulation, les variables à fixer sont les composantes w et b. Le vecteur ***w*** possède un nombre de composantes égal à la dimension de l'espace d'entrée. Généralement dans ce type de cas, on résout la forme duale du problème. Le passage du problème primal au dual introduit trois principes mathématiques qui sont : principe de Fermat, principe de Lagrange et principe de Kuhn-Tucker.

Nous devons faire rentrer les contraintes dans la fonction objective et de pondérer chacune d'entre elles par une variable duale (appliquer le principe de Lagrange).

$$L(w,b,a) = \frac{1}{2}w^2 + \sum_{i=1}^{n} \alpha_i [<w,x_i> + b)-1] \qquad \textbf{I13}$$

Notons que L doit être minimisé par rapport aux variables primales w_i et b

et maximisé par rapport aux variables duales α_i.

Le point selle (minimal par rapport à une variable, maximal par rapport à l'autre) doit donc satisfaire les conditions nécessaires de stationnarité (annule sa dérivé) qui correspondent aux conditions Karush Kuhn et Tucker (KKT) et de Fermat, nous trouvons:

$$\frac{6L(w,b,\alpha)}{6w} = 0 \qquad \textbf{I14}$$

$$\frac{6L(w,b,\alpha)}{6b} = 0 \qquad \textbf{I15}$$

Ce qui nous permet
d'obtenir

$$W = \sum_{i=0}^{n} \alpha_i y_i X_i \qquad \textbf{I17}$$

$$\sum_{i=0}^{n} \alpha_i y_i = 0 \qquad \textbf{I16}$$

Remarquons qu'avec cette formulation, on peut calculer w en fixant seulement n paramètres. L'idée va donc être de formuler un problème dual dans lequel w est remplacé par sa nouvelle formulation. De cette façon, le nombre de paramètres à fixer est relatif au nombre d'exemples de l'échantillon d'apprentissage et non plus à la dimension de l'espace d'entrée.

Pour se faire, nous substituons (II.16) et (II.17) dans le Lagrangien (II.13), nous obtenons le problème dual équivalent suivant :

$$PQ2 \begin{cases} \underbrace{max}_{\alpha} \sum_{i=1}^{n} \alpha i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \\ SC \qquad \sum_{i=0}^{n} \alpha_i y_i = 0 \\ \qquad\quad \alpha_i \geq 0 \end{cases} \qquad \textbf{I18}$$

Ce dernier problème peut être résolu en utilisant des méthodes standards de programmation quadratiques. Une fois la solution optimale α^* du problème obtenue,le vecteur de poids de l'hyperplan à marge maximale recherché s'écrit :

$$w^* = \sum_{i=1}^{N_{vs}} \alpha_i^* y_i^* x_i^* \qquad \textbf{N_{vs} nombre de vecteurs support} \qquad \textbf{I19}$$

Avec N_{vs}= nombre de vecteurs support

Comme le paramètre b ne figure pas dans le problème dual, sa valeur optimale b^* peut être dérivée à partir des contraintes primales, soit donc :

$$b^* = -\frac{max_{yi=-1}(<w^*,x_i>)+min_{yi=1}(<w^*,x_i>)}{2} \qquad \textbf{I20}$$

Nous avons à présent tous les éléments nécessaires pour exprimer la fonction de décision de notre classificateur linéaire :

$$h(x) = sign(\sum_{i=1}^{nvs} \alpha i^* y_i^* \langle x, x_i^* \rangle + b^*) \qquad \textbf{I21}$$

Notons qu'un grand nombre de termes de cette somme est nul. En effet,

seuls les α_i^* correspondants aux exemples se trouvant sur les hyperplans canoniques (sur la contrainte) sont non nuls. Ces exemples sont appelés Supports Vectors (**SV**). On peut les voir comme les représentants de leurs catégories car si l'échantillon d'apprentissage n'était constitué que des **SV**, l'hyperplan optimal que l'on trouverait serait identique.

b- Marge souple ((∃ $\mathcal{E}_i \neq 0$) et C ≥ 0 dans (PQ1))

Nous considérons ici le cas où certains exemples sont mal classés par l'hyperplan optimal. Cela peut résulter du bruit dans les données. Pour résoudre ce problème, Cortes et Vapnik en 1995 ont introduit la notion de « marge souple » (soft margin) qui correspond toujours à la recherche d'un hyperplan de marge optimale, mais avec une règle d'exception qui autorise que quelques exemples soient à une distance plus faible de l'hyperplan que la marge correspondante.

Soit si = $1 - y_i.h(x_i)$ un indice mesurant l'importance de pénétration de l'exemple xi dans la zone définie par l'hyperplan H de marge géométrique d, $\mathcal{E}_i \neq 0$ pour $h(x_i) < 1$. Cette variable est appelée variable ressort (slack variable). S$\mathcal{E}_i > 1$i, l'exemple n'est pas du bon côté de l'hyperplan relativement à sa classe (exemple : yi = 1, h(xi) = −1 et $\mathcal{E}_i = 2 > 1$).

L'idée de la marge souple est de rechercher l'hyperplan de marge optimale pénalisée par l'importance des variables ressorts. Le terme de marge souple vient du fait que l'on peut considérer que les exemples pour les quels $\mathcal{E}_i > 0$, ont une marge géométrique réduite de d(1 − $\mathcal{E}_i$). Le terme de pénalisation est de la forme C $\sum\mathcal{E}_i$ avec C une constante qui permet de définir l'importance de la pénalisation.

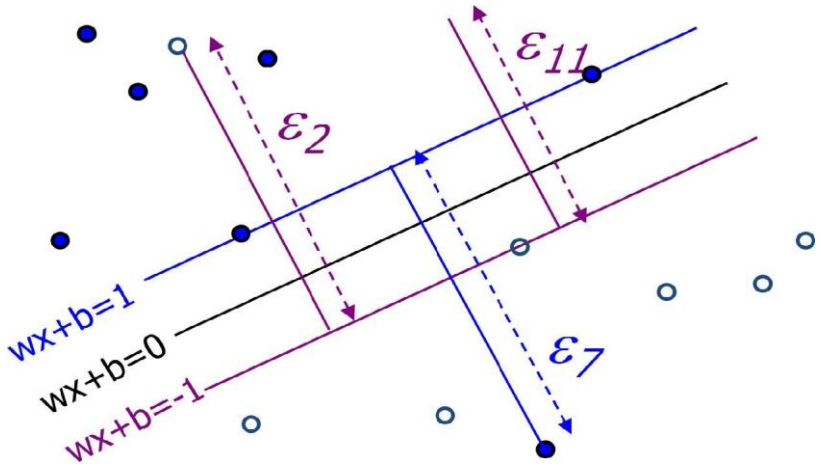


Figure 17 : Marge souple et variable élastique εi

Le paramètre ***C*** est défini par l'utilisateur. Il peut être interprété comme une tolérance au bruit du classifieur : pour de grandes valeurs de ***C***, seules de très faibles valeurs de ç sont autorisées, et par conséquent, le nombre de points mal classés sera très faible (données faiblement bruitées). Si ***C*** est petit, s peut devenir très grand, et on autorise alors bien plus d'erreurs de classification (données fortement bruitées).

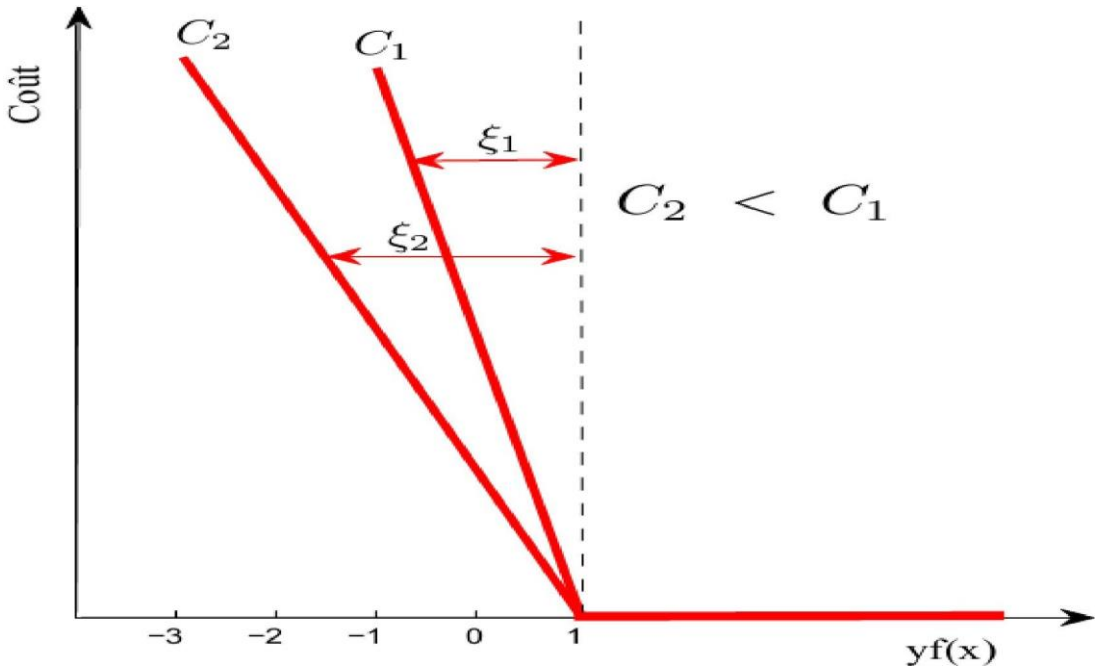


Figure 18 : Représentation du compromis entre la tolérance C et la variable élastique εi.

En suivant la même démarche du Lagrangien que précédemment, nous aboutissons à

la forme duale suivante :

$$L(w,b,\alpha,\varepsilon,\beta) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\varepsilon_i - \sum_{i=1}^{n}\alpha_i[y_i(\langle w,x_i\rangle + b) - 1 + \varepsilon_i] - \sum_{i=1}^{n}\beta_i.\varepsilon_i \quad \textbf{I22}$$

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{n}\alpha_i.y_i.x_i \quad \textbf{I23}$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{n}\alpha_i.y_i = 0 \quad \textbf{I24}$$

$$\frac{\partial L}{\partial \varepsilon} = C - \alpha_i - \beta_i = 0 \quad \textbf{I25}$$

Ce qui nous permet d'obtenir :

$$w = \sum_{i=1}^{n}\alpha_i.y_i.x_i \quad \textbf{I26}$$

$$\sum_{i=1}^{n}\alpha_i.y_i = 0 \quad \textbf{I27}$$

$$\alpha_i = C - \beta_i \qquad 0 \le \alpha_i \le C \quad \textbf{I28}$$

Ces conditions sont injectées dans (II.22) pour passer au problème dual :

$$\textbf{PQ3}\begin{cases} max\ \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\alpha_i\alpha_j y_i y_j\langle x_i.x_j\rangle & \textbf{I29} \\ SC\ \sum_{i=1}^{n}\alpha_i y_i = 0 & \textbf{I30} \\ 0 \le \alpha \le C \end{cases}$$

Pour résoudre ce problème ***(PQ3)*** il existe plusieurs algorithme tels que le SMO (Sequentiel Minimisation Optimisation) [11], SVM light et Simple SVM afin de trouver les vecteurs de support.

3.2.2. Cas non linéairement séparable

Précédemment, nous avons décrit le principe des SVM dans le cas où les données sont linéairement séparables. Cependant, dans la plupart des problèmes réels, ce n'est pas toujours le cas et il est donc nécessaire de contourner ce problème (difficile de séparer n'importe quel jeu de données par un simple hyperplan). Si par exemple les données des deux classes se chevauchent sévèrement, aucun hyperplan séparateur ne sera satisfaisant.

L'idée est de projeter les points d'apprentissage dans un espace d'Hilbert H de dimension plus élevée dans lequel les données transformées devienne linéairement séparable et ce grâce à une fonction Φ non-linéaire choisie a priori et d'appliquer la même méthode d'optimisation de la marge dans cet espace. L'espace ainsi obtenu est appelé espace des caractéristiques ou aussi espace transformé.

Le principe revient donc à résoudre les problèmes ***PQ2*** et ***PQ3*** dans l'espace H, en remplaçant $\{xi.\ X\}$ par $\{\ \Phi(xi).\Phi(xj)\}$.

Le problème quadratique obtenu peut s'écrire comme suit :

$$PQ4 \begin{cases} max\ \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \Phi(x_i).\Phi(x_j)\rangle & \text{I31} \\ SC\ \sum_{i=1}^{n} \alpha_i y_i = 0 & \text{I32} \\ 0 \le \alpha_i \le C \end{cases}$$

a-Astuce de noyau

Vu que les données apparaissent dans tous les calculs uniquement sous forme de produits scalaires { $\Phi(x_i).\Phi(x_j)$ }, il suffit de faire appel à une fonction noyau $K(x_i, x_j)$ qui permet ce calcul $K(x_i, x_j) = \{ \Phi(x_i).\Phi(x_j)\}$ et qui satisfait la condition de Mercer.

Condition de Mercer

On dit d'une fonction est un noyau si et seulement si la condition suivante est vérifiée:

$G = K(x_i, x_j)_{i,j=1}^{n}$ est défini positive autrement dit elle vérifie les trois propriétés fondamentales du produit scalaire :

- Positivité : $K(x_i, x_j) \geq 0$
- Symétrie : $K(x_i, x_j) = K(x_j, x_i)$
- Inégalité de Cauchy-Shwartz : $K(x_i, x_j) \leq \|x_i\|.\|x_j\|$

G est une matrice contenant les similarités entre tous les exemples de l'ensemble d'apprentissage appelée matrice de Gram, elle a une importance cruciale dans les algorithmes à noyaux car c'est elle qui définit la complexité numérique de l'apprentissage ; pour le problème de la classification SVM, elle permet de définir la partie quadratique de la forme quadratique à optimiser et elle contient aussi toutes les informations sur les données d'apprentissage et la fonction K.

Parmi toutes les fonctions noyaux utilisées pour répondre aux besoins des SVM, on cite les plus utilisées :

Le noyau linéaire : est un simple produit scalaire : $K(x_i, x_j) = \{x_i, x_j\}$ **I33**

Le noyau polynomial : permet de représenter des frontières de décision par des polynômes de degré d : $K(x_i, x_j) = (a\times\{x_i, x_j\}+ b)^d$ **I34**

La dimension de l'espace transformé induit par un noyau polynomial est de l'ordre

$\frac{(p+d)!}{p!d!}$, où p est la dimension de l'espace de départ.

Le noyau gaussien ou RBF (Radial Basis Function) : qui a la forme suivante :

$$exp = (-\frac{\|xi - xj\|^2}{2\sigma^2}) \qquad \textbf{I34}$$

Le paramètre σ permet de régler la largeur de la gaussienne. En prenant un σ grand, la similarité d'un exemple par rapport à ceux qui l'entourent sera assez élevée, alors qu'on prenant un σ tendant vers 0, l'exemple ne sera similaire à aucun autre.

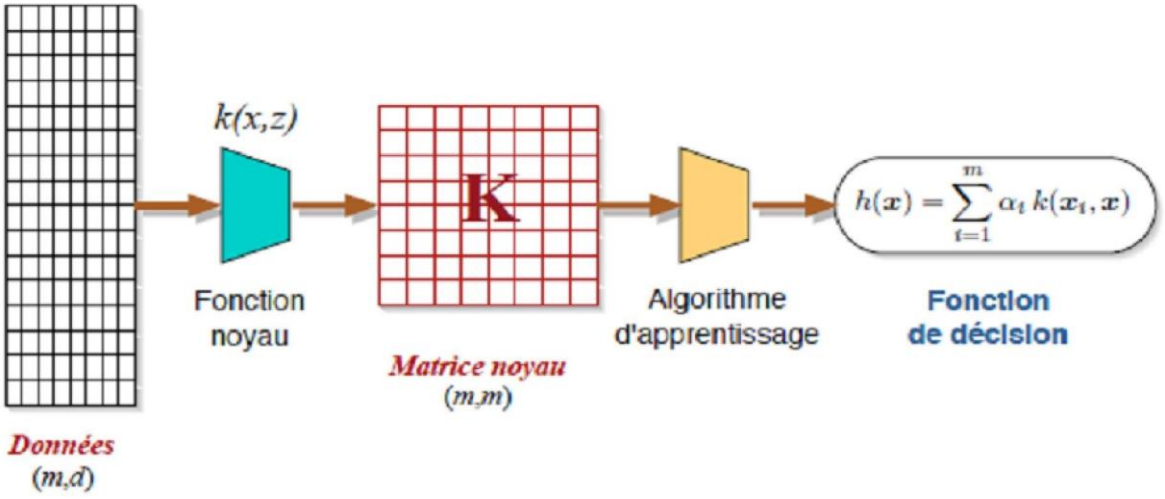


Figure 19 : Chaîne de traitements génériques d'une méthode à noyau.

3.3.Le choix des paramètres optimaux

La réalisation d'un programme d'apprentissage par SVM se ramène à résoudre un problème d'optimisation impliquant un système de résolution dans un espace de dimension conséquente. L'utilisation de ces programmes revient surtout à sélectionner une bonne famille de fonctions noyau et à régler les paramètres de ces fonctions. Ces choix sont le plus souvent faits par une technique de validation croisée, dans laquelle on estime la performance du système en la mesurant sur des exemples n'ayant pas été utilisés en cours d'apprentissage.

L'idée est de chercher les paramètres permettant d'obtenir la performance maximale. Si la mise en œuvre d'un algorithme de SVM est en général peu coûteuse en temps, il faut cependant compter que la recherche des meilleurs paramètres peut requérir des phases de test assez longues.

4.Extension des SVM

La plupart des problèmes ne se contentent pas de deux classes de données. Il existe plusieurs méthodes pour faire la classification multi-classes.

La première méthode est appelé Un-Contre-Tous. C'est une approche étendant la notion de marge aux cas multi-classes. Cette formulation intéressante permet de poser un problème d'optimisation unique. Le problème fait intervenir N fonctions de décision.

La deuxième méthode est une méthode dite Un-contre-Un. Au lieu d'apprendre N fonctions de décisions, ici chaque classe est discriminée d'une autre.

4.1.Approche Un-contre-Tous(1vsR)

L'idée de cette stratégie est de construire autant de classificateurs que de classes. Ainsi, durant l'apprentissage, tous les exemples appartenant à la classe considérée sont étiquetés positivement (+1) et tous les exemples n'appartenant pas à la classe sont étiquetés négativement (-1).

Un hyperplan H_k est défini pour chaque classe k par la fonction de décision suivante :

$$H_k(x) = sign(\langle w_k, x\rangle + b_k) \qquad \textbf{I36}$$

$$= \begin{cases} +1, si\ H_k(x) > 0 \\ -1, sinon \end{cases}$$

$$k^* = Arg_{1\leq k\leq K}\, Max(H_k(x)) \qquad \textbf{I37}$$

Si une seule valeur $H_k(x)$ est égale à 1 et toutes les autres sont égales à -1, on conclut que x appartient à la classe k. Or, il est possible que plusieurs sorties soient positives pour un exemple de test donné. Ceci est particulièrement le cas des données ambiguës situées près des frontières de séparation des classes. On utilise dans ce cas un vote majoritaire pour attribuer l'exemple x à la classe k.

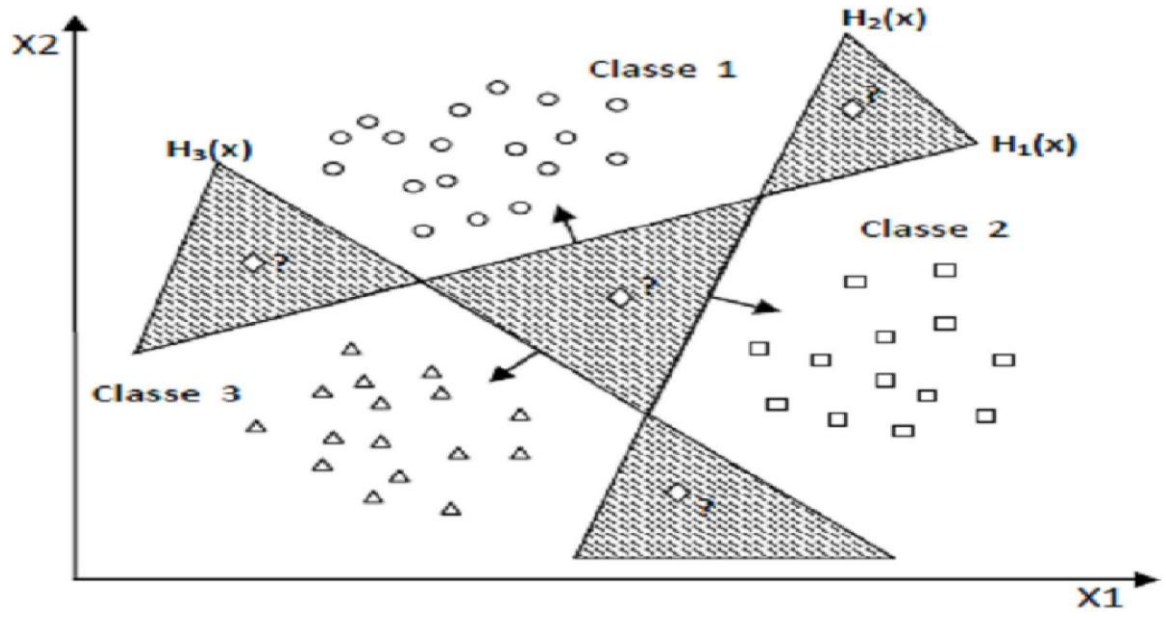


Figure 20 : Approche un contre tous.

Cette méthode est critique à cause de son asymétrie, puisque chaque hyperplan est entrainé sur un nombre d'exemples négatifs beaucoup plus important

que le nombre d'exemples positifs. La méthode un contre un est une méthode symétrique qui corrige ce problème.

4.2.Approche Un-contre-Un(1vs1)

L'approche Un-Contre-Un est un cas spécial des méthodes de décomposition proposées par Dietrich et al. [13] pour résoudre des problèmes à plusieurs classes. Cette approche requiert la construction de K (K − 1)/2 SVM chacun séparant un couple de classes (i, j) parmi ceux existants.

Pendant la classification, un vecteur d'entrée x est présenté à l'ensemble des classificateurs construits. La sortie de chaque SVM fournit un vote partiel concernant uniquement le couple de classes (wi,wj). En considérant que chaque SVM calcule un estimé $\hat{P}_{ij}$ de la probabilité :

$$p_{ij} = P(x \in w_i | x, x \in w_i \cup w_j) \qquad \textbf{I38}$$

Alors la règle de classification la plus simple peut s'écrire :

$$\operatorname{argmax}_{1<i\leq k} \sum_{j\neq i} [\hat{P}_{ij} > 0.5] \qquad \textbf{I39}$$

L'opérateur η est défini :

$$[\eta] = \begin{cases} 1 \text{ si } \eta \text{ est vrai} \\ 0 \text{ sinon} \end{cases} \qquad \textbf{I40}$$

Cette combinaison considère que les sorties des SVM sont des valeurs binaires de 1,0.

Une autre approche de reconstruction pourrait tirer avantage de l'information de confiance associée à chacune des sorties p_{ij}. Dans l'hypothèse que ces valeurs représentent des probabilités, il est possible d'estimer une approximation $\hat{e}_i$ de la probabilité à posteriori : $\hat{P}_{ij} = P(x \in w_i | x)$ **I41**

En considérant la matrice carrée $\hat{P}$ avec les entrées $\hat{P}_{ij}$ tels que $(i, j)_{i,j=1,\dots k.}$ avec $\hat{P}_{ij}=1-\hat{P}_{ij}$. Les valeurs de $\hat{e}_i$ peuvent être calculées par :

$$\hat{P}_i = \frac{2}{K(K-1)} \sum_{i \neq j} \hat{P}_{ij} \qquad \textbf{I42}$$

Et la règle de décision s'écrit :

$$\arg\max_{1 \leq i \leq K} \hat{P}_i \qquad \textbf{I43}$$

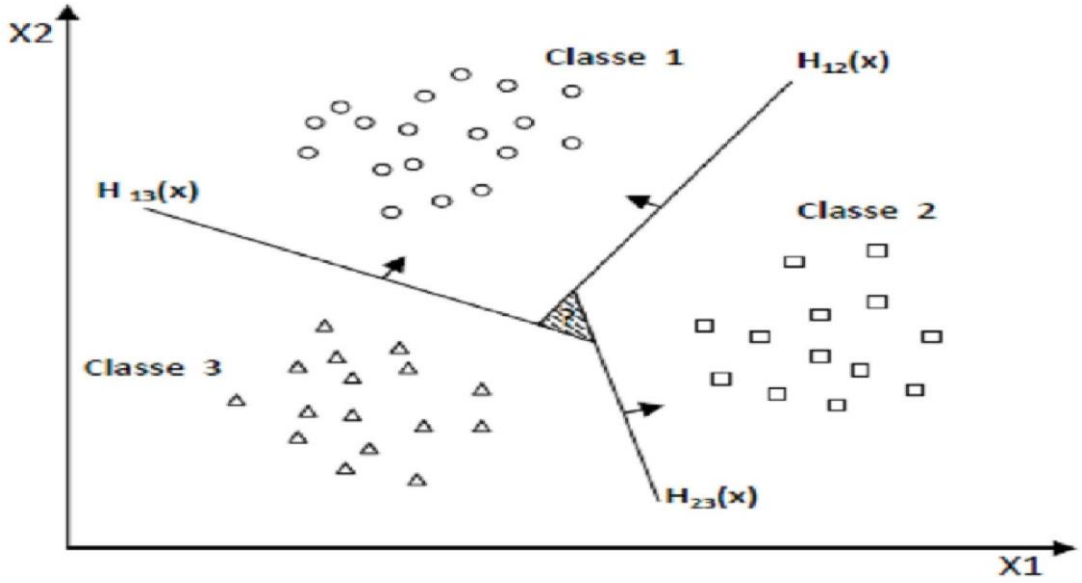


Figure 21 : Approche un contre un. [10]

Conclusion

Dans ce chapitre, nous avons présenté de manière simple et complète le concept de système d'apprentissage introduit par Vladimir Vapnik, les « Support Vector Machine». Nous avons donné une vision générale et une vision purement mathématiques des SVM. Cette méthode de classification est basée sur la recherche d'un hyperplan qui permet de séparer au mieux des ensembles de données. Nous avons exposé les cas linéairement séparable et les cas non linéairement séparables qui nécessitent l'utilisation de fonction noyau pour changer d'espace.

Chapitre 05 :

Conception et implementation de l'appliction en python

Chapitre V. Conception et implémentation de l'application en python.

Introduction

Notre choix du langage de programmation s'est porté sur le langage python, et cela parce qu'il est un langage orienté objet simple et il possède une riche bibliothèque de classes comprenant des fonctions diverses, ainsi que beaucoup de fonctionnalités qui peuvent être utilisé pour développer des applications diverses surtout les bibliothèques de apprentissage automatique et simple pour utilisation.

Classification des pourriels[9]

Un classificateur SVM sera formé pour déterminer si un email donné x est un spam (y=1) ou non (y=0). En particulier, chaque email doit être converti en un vecteur caractéristique x∈Rn

L'ensemble de données est basé sur un sous-ensemble du SpamAssassin Public Corpus (Corpus de courrier public de SpamAssassin. Ceci est une sélection de courrier messages, utilisables pour tester les systèmes de détection anti-spam) et seul le corps de l'email sera utilisé (à l'exclusion des en-têtes d'e-mail).

1- Prétraitement des emails

Exemple d'email :

```
> Anyone knows how much it costs to host a web portal ?
>
Well, it depends on how many visitors you're expecting.
This can be anywhere from less than 10 bucks a month to
a couple of $100.
You should checkout http://www.rackspace.com/ or
perhaps Amazon EC2
if you're running something big..


To unsubscribe yourself from this mailing list, send an
email to:
groupname-unsubscribe@egroups.com
```

Avant de commencer une tâche d'apprentissage automatique, il est généralement utile de jeter un œil à des exemples de l'ensemble de données. L'exemple d'email contient une URL, une adresse email (à la fin), des chiffres et des montants en dollars. Alors que de nombreux emails contiendraient des types d'entités similaires (par exemple, des nombres, d'autres URL ou d'autres adresses e-mail), les entités spécifiques (par exemple, l'URL spécifique ou le montant en

[9] Pourriels: Contraction de pourri et de courriel, ce terme désigne les spam arrivant des les boîtes mails en grande quantité.

dollars spécifique) seront différentes dans presque chaque e-mail. Par conséquent, une méthode souvent utilisée dans le traitement des emails consiste à « normaliser » ces valeurs, afin que toutes les URL soient traitées de la même manière, que tous les nombres soient traités de la même manière, etc. Par exemple, nous pourrions remplacer chaque URL de l'email par la chaîne unique "httpaddr" pour indiquer qu'une URL était présente.

Cela a pour effet de laisser le classificateur de spam prendre une décision de classification en fonction de la présence d'une URL plutôt que de la présence d'une URL spécifique. Cela améliore généralement les performances d'un classificateur de spam, car les spammeurs randomisent souvent les URL, et donc les chances de voir à nouveau une URL particulière dans un nouveau spam sont très faibles.

Dans process_email.py, les étapes de prétraitement et de normalisation des emails suivantes ont été mises en œuvre :

Minuscules : l'intégralité de l'email est convertie en minuscules, de sorte que la majuscule est ignorée (par exemple, IndIcaTE est traité de la même manière que Indiquer).

Suppression du HTML : toutes les balises HTML sont supprimées des emails. De nombreux emails sont souvent au format HTML ; nous supprimons toutes les balises HTML, de sorte que seul le contenu reste.

Normalisation des URL : toutes les URL sont remplacées par le texte "httpaddr".

Normalisation des adresses email : toutes les adresses email sont remplacées par le texte "emailaddr".

Numéros de normalisation : tous les numéros sont remplacés par le texte "numéro".

Normalisation des dollars : tous les signes dollar ($) sont remplacés par le texte "dollar".

Racine de mots : les mots sont réduits à leur forme radicale. Par exemple, "includ", "discounts", "discounted" et "discounting" sont tous remplacés par "discount". Parfois, le Stemmer supprime en fait des caractères supplémentaires à la fin, de

sorte que "include", "includes", "included" et "including" sont tous remplacés par "includ".

Suppression des non-mots : Les non-mots et la ponctuation ont été supprimés. Tous les espaces blancs (tabulations, nouvelles lignes, espaces) ont tous été réduits à un seul caractère d'espacement.

Le résultat de ces étapes de prétraitement ressemble au paragraphe suivant :

```
Python 2.7.18 (v2.7.18:8d21aa21f2, Apr 20 2020, 13:25:05) [MSC v.1500 64 bit (AMD64
)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: D:\_app_\Spam Classification\_app_\ex6_spam.py ==========
Preprocessing sample email (emailSample1.txt)...
anyon know how much it cost to host a web portal well it depend on how mani visitor
 you re expect thi can be anywher from less than number buck a month to a coupl of
dollarnumb you should checkout httpaddr or perhap amazon ecnumb if your run someth
big to unsubscrib yourself from thi mail list send an email emailaddr
```

Alors que le prétraitement a laissé des fragments de mots et des non-mots, cette forme s'avère beaucoup plus facile à utiliser pour effectuer l'extraction de caractéristiques.

Liste de vocabulaire

Après le prétraitement des emails, il existe une liste de mots pour chaque e-mail. L'étape suivante consiste à choisir quels mots seront utilisés dans le classificateur et lesquels seront laissés de côté.

Pour des raisons de simplicité, seuls les mots les plus fréquents dans l'ensemble de mots considéré (la liste de vocabulaire) ont été choisis. Étant donné que les mots qui apparaissent rarement dans l'ensemble d'apprentissage ne se trouvent que dans quelques emails, ils peuvent amener le modèle à sur adapter l'ensemble d'apprentissage. La liste complète du vocabulaire se trouve dans le fichier vocab.txt. La liste de vocabulaire a été sélectionnée en choisissant tous les mots qui apparaissent au moins 100 fois dans le corpus de spam, résultant en une liste de 1899 mots. En pratique, une liste de vocabulaire avec environ

10000 à 50 000 mots est souvent utilisée.

Compte tenu de la liste de vocabulaire, chaque mot peut désormais être mappé dans les emails prétraités dans une liste d'index de mots qui contient l'index

du mot dans la liste de vocabulaire. Par exemple, dans l'exemple d'e-mail, le mot "anyone" a d'abord été normalisé en "anyon", puis mis en correspondance avec l'index 86 de la liste de vocabulaire.

Le code dans process_email.py effectue ce mappage. Dans le code, une chaîne donnée str qui est un seul mot de l'email traité est recherchée dans la liste de vocabulaire vocabList.txt Si le mot existe, l'index du mot est ajouté dans la variable word_indices. Si le mot n'existe pas, et n'est donc pas dans le vocabulaire, le mot peut être sauté.

```
with open('emailSample1.txt', 'r') as email:

file_contents = email.read()

file_contents
```

```
                                                                    Word Indices:
 [85, 915, 793, 1076, 882, 369, 1698, 789, 1821, 1830, 882, 430, 1170, 793, 1001, 1
892, 1363, 591, 1675, 237, 161, 88, 687, 944, 1662, 1119, 1061, 1698, 374, 1161, 47
8, 1892, 1509, 798, 1181, 1236, 809, 1894, 1439, 1546, 180, 1698, 1757, 1895, 687,
1675, 991, 960, 1476, 70, 529, 530]
```

2- Extraction de fonctionnalités à partir d'emails

L'extraction de caractéristiques qui convertit chaque email en un vecteur dans Rn

devrait être mis en œuvre. Pour cela, n=# mots dans la liste de vocabulaire seront utilisés. Plus précisément, la caractéristique xi∈{0,1} pour un email correspond au fait que le $i^{ème}$ mot du dictionnaire se trouve dans l'email. C'est-à-dire xi=1 si le $i^{ème}$ mot est dans l'email et xi=0

si le $i^{ème}$ mot n'est pas présent dans l'email.

Ainsi, pour un email typique, cette fonctionnalité ressemblerait à :

$$x = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \in \mathbb{R}^n$$

Le code dans email_features.py génère un vecteur de caractéristiques pour un e-mail, étant donné les indices de mot. En exécutant le code sur l'exemple de courrier électronique, le vecteur de caractéristiques aura une longueur de 1899 et 43 entrées non nulles.

3- Formation SVM pour la classification des spams

Ensuite, un ensemble de données d'entraînement prétraité sera chargé et il sera utilisé pour entraîner un classificateur SVM. spamTrain.mat contient 4000 exemples de formation de courrier indésirable et non-spam, tandis que spamTest.mat contient 1000 exemples de test. Chaque email d'origine a été traité à l'aide des fonctions process_email et email_features.py et converti en un vecteur x(i)∈R1899.

```
Extracting features from sample email (emailSample1.txt)...
Length of feature vector: 1899
Number of non-zero entries: 45
Training Linear SVM (Spam Classification)...
Training Accuracy: 99.97500000000001
```

4- Tester la classification des spams

Après le chargement de l'ensemble de données, un SVM sera formé pour classer entre spam (y=1) et les emails non-spam (y=0). Une fois l'apprentissage terminé, le classificateur obtient une précision d'apprentissage d'environ 99,8 % et une précision de test d'environ 98,9 %.

```
Evaluating the trained Linear SVM on a test set...
Test Accuracy: 99.2
```

5- Principaux prédicteurs de spam

Pour mieux comprendre le fonctionnement du classificateur de spam, nous pouvons inspecter les paramètres pour voir quels mots le classificateur pense être les plus prédictifs du spam. Ensuite, les paramètres avec les plus grandes valeurs positives dans le classificateur seront trouvés et les mots correspondants seront affichés.

```
Top predictors of spam:
our              (0.421665)
remov            (0.387173)
click            (0.387060)
basenumb         (0.346617)
guarante         (0.341686)
visit            (0.303028)
bodi             (0.263524)
will             (0.244394)
numberb          (0.238795)
price            (0.234199)
dollar           (0.232315)
nbsp             (0.227081)
below            (0.223199)
lo               (0.219994)
most             (0.214548)
```

Ainsi, si un email contient des mots tels que « garantir », « retirer », « dollar » et « prix », il est susceptible d'être classé comme spam.

6-Essayez un emails

Pour exécuter le classificateur de spam sur spamSample1.txt :

```
Do You Want To Make $1000 Or More Per Week?
If you are a motivated and qualified individual - I
will personally demonstrate to you a system that will
make you $1,000 per week or more! This is NOT mlm.
Call our 24 hour pre-recorded number to get the
details.
000-456-789
I need people who want to make serious money.  Make
the call and get the facts.
Invest 2 minutes in yourself now!
000-456-789
Looking forward to your call and I will introduce you
to people like yourself who
are currently making $10,000 plus per week!
000-456-789
3484lJGv6-241lEaN9080lRmS6-271WxHo7524qiyT5-438rjUv5615hQcf0-
662eiDB9057dMtVl72
```

```
do you want to make dollarnumb or more per week if you are a motiv and qualifi individu i will person demonstr to you a system that w
ill make you dollarnumb number per week or more thi is not mlm call our number hour pre record number to get the detail number number
number i need peopl who want to make seriou money make the call and get the fact invest number minut in yourself now number number nu
mber look forward to your call and i will introduc you to peopl like yourself whoar current make dollarnumb number plu per week numbe
r number numberljgvnumb numberleannumberlrmsnumb numberwxhonumberqiytnumb numberrjuvnumberhqcfnumb numbereidbnumberdmtvlnumb Processe
d spamSample1.txt
Spam Classification: [1]
(1 indicates spam, 0 indicates not spam)
>>>
```

Essayer des emails depuis Enron[10]

emails	**Notre classification**	**Enron Corpus classification**
Subject: re [8] : dear friend - size = 1 > order confirmation . your order should be shipped by january , via fedex . your federal express tracking number is 45954036 . thank you for registering . your userid is : 56075519 learn to make a fortune with ebay ! complete turnkey system software - videos - turorials clk here for information clilings	Spam	Spam
Subject: meter 1431 - nov 1999 daren - could you please resolve this issue for howard ? i will be out of the office the next two days . when this is done , please let george know . thanks .aimee	Ham	Ham
subject : miscellaneous sorry i ' m just now getting back to you . here are some answers to your questions . waskom is a field in east texas (harrison county) where we purchase gas from pennzenergy at the bryson c . p . and jeter # 2 wells . the gas is termed up to new	Ham	Ham

Tableau 1 : exemple des email depuis enron tester avec l'application.

[10] Le corpus Enron est une base de données de plus de 600 000 e-mails générés par 158 employés d'Enron Corporation au cours des années qui ont précédé l'effondrement de l'entreprise en décembre 2001. Le corpus a été généré à partir des serveurs de messagerie d'Enron par la Federal Energy Regulatory Commission (FERC) au cours de sa enquête ultérieure.

Conclusion

générale

CONCLUSION GENERALE

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 significativement le taux de détection de spam, par la progression de classification des emails en spam et légitime.

À l'heure actuelle, les techniques de détection de spam à base d'apprentissage sont loin d'être performants à 100%, car ces dernières ne traitent pas de la sémantique. Donc, il est très important de continuer à progresser d'une part dans le domaine de traitement linguistique de textes, afin d'arriver à une représentation textuelle manipulable par l'algorithme de classification utilisé en gardant la sémantique du texte. Et d'autre part, utiliser des algorithmes de classification performants pour la classification des courriels.

Récemment, les chercheurs ont examiné l'utilisation de la sémantique dans le détection des spams en représentant des emails avec un nouveau modèle vectoriel qui utilise une ontologie pour représenter les différentes relations entre les termes et, de cette manière, il offre un modèle plus riche. Sur la base de cette représentation, ils appliquent plusieurs classifieurs bien connus (SVM, NB, K-ppv et AD) et montrent que la méthode proposée permet de détecter la sémantique interne des messages et que cette approche donne des pourcentages élevés de détection de spam.

L'objectif de notre travail se dirigeait vers le développement d'une approche

d'apprentissage automatique, afin d'améliorer la performance du système de détection avec SVM. Malgré les performances de ce système, il est intéressant de continuer le travail sur d'autres corpus, et appliquer une combinaison avec d'autres classifieurs tels que : naïve bayes, les réseaux de neurones, etc.

Références :

Référence

[1] Gherabi Charaf Eddine, “Détection des spams se basant sur les techniques de classification”, Mémoire de Master Universite Mohamed boudiaf- M’sila, 2018.
[2] OECD, "Boîte à outils anti-spam de l'OCDE Politiques et mesures recommandées", 2006.
[3] Boumediene Hassan, “Algorithme de boosting et meta-heuristique base sur la pso pour la detection et le Filtrage De Spam”, Mémoire de Master Universite Tahar Moulay- Saida, 2013.
[4] Hossain, Md Forhad, "Fake Review Detection using Data Mining", MSU Graduate Theses, Missouri State University, https://bearworks.missouristate.edu, 2019.
[5] SiteWeb ICHI.PRO, “Filtrage des courriers électroniques anti-spam non spam à l'aide de l'apprentissage automatique”, https://ichi.pro/, 2020.
[6] Abdul Jabbar Saleh, Asif Karim, “An Intelligent Spam Detection Model Based on Artificial Immune. System”, Charles Darwin University Australia, https://pdfs.semanticscholar.org/, 2019.
[7] Rami Mustafa A. Mohammad, “A lifelong spam emails classification model”, Article, Imam Abdulrahman Bin Faisal University, Dammam, Saudi Arabia, https://www.emerald.com/, 2019.
[8] Ajay Rastogi, Monica Mehrotra, Syed Shafat Ali, “Effective Opinion Spam Detection: A Study on Review Metadata Versus Content”, Jamia Millia Islamia, India, http://manu47.magtech.com.cn/, 2020.
[9] Shah Nazir, Habib Ullah Khan, “Spam Detection Approach for Secure Mobile Message Communication Using Machine Learning Algorithms”,University, Xianyang, China, https://www.hindawi.com/, 2020.
[10] Jiayong Liu, Yu Su, Shun Lv, Cheng Huang, “Detecting Web Spam Based on Novel Features from Web Page Source Code”, Sichuan University, 2020.
[11] Redouane LEKHAL, “Application des SVM pour la reconnaissance d’extrasystoles.”, Mémoire de Fin d’Etudes de MASTER ACADIMIQUE, UNIVERSITE MOULOUD MAMMERI DE TIZIOUZOU, 2015.