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*
جامعة غرداية
*Université de Ghardaïa*
كلية العلوم والتكنولوجيا
*Faculté des Sciences et de Technologie*
قسم الرياضيات و الإعلام الآلـي
*Département des Mathématiques et d'informatique*

*N° d'enregistrement*
/...../...../...../...../.....

Mémoire de fin d'études

Présenté pour l'obtention du diplôme de MASTER

**En :** ***Informatique***

**Option** : Systèmes Intelligents pour l'Extraction de Connaissances (SIEC)

## Thème

# Détection des utilisateurs indésirables dans Twitter en utilisant les techniques PSO-EHO

***Réalisé par :***

*RECIOUI Makhlouf & YAGOUB Menaa*

**Soutenu le 24/06/2023, devant le jury composé de :**

| | | | |
|---|---|---|---|
| M. Nacira BRAHIM | MCB | Univ. Ghardaïa | - Président |
| M. A. BOUCHEKOUF | MCB | Univ. Ghardaïa | - Examinateur |
| M. Ahmed SAIDI | MCB | Univ. Ghardaïa | - Examinateur |
| M. Abdelkader BOUHANI | MCB | Univ. Ghardaïa | - Encadrant |

***Promotion : 2022/2023***

# Dédicace

Je dédie ce travail, À mes chers parents, qui ont toujours été là pour me Soutenir et m'encourager.

À mes frères,

À tous les membres de ma famille,

À tous mes chers amis, A toutes les personnes qui m'ont soutenue dans la Réalisation de ce travail.

Merci.

- YAGOUB Menaa

# Dédicace

Je dédie ce travail, À mes chers parents, qui
ont toujours été là pour me
Soutenir et m'encourager.
À ma chère Epouse et mes chères enfants,
À mes frères,
À tous les membres de ma famille,
À tous mes chers amis, A toutes les personnes
qui m'ont soutenue dans la
Réalisation de ce travail.
Merci.

- RECIOUI Makhlouf

# ملخص

تجذب الشبكات الاجتماعية ملايين المستخدمين في جميع أنحاء العالم من مختلف الأعمار والفئات، مما يؤهلها لتصبح منصات مستهدفة من قبل المستخدمين المزيفين لنشر قدر هائل من المعلومات الخاطئة و المزيفة.

تويتر على سبيل المثال، أصبح أحد المنصات التي تجذب المحتالين والمستخدمين المخادعين الذين ينشرون معلومات غير صحيحة للمستخدمين من خلال هويات مزيفة، مما أدى إلى انتشار المحتوى الضار.

في الآونة الأخيرة تم اقتراح العديد من التقنيات لضمان أمن هذه الشبكة ولكن كل هذه التقنيات لم تكن قادرة على وضع حد لهذا التحدي، ومن بين هذه التقنيات نجد خوارزميات التعلم الآلي و المستوحاة من الطبيعة، وبالطبع هناك العديد من الطرق الأخرى. و التي من ضمنها خوارزمية SVM.

يهدف عملنا إلى دراسة ما إذا كان تقنية تحسين الخصائص, من خلال الخوارزميات المستوحاة من الطبيعة مثل PSO و EHO يمكن من تحسين النتائج التي تم الحصول عليها. هدف هذا العمل إجراء مقارنة على مستوى طريقة SVM ودمجها مع PSO ثم مع EHO.

ومن ثم إجراء مراجعة للتقنيات المستخدمة لاكتشاف الحسابات المزيفة على تويتر مع محاولة تحسين طريقة اكتشاف بناءًا على خوارزميات مستوحاة من الطبيعة مثل تحسين سرب الجسيمات (PSO) وتحسين رعي االفيلة (EHO).

إذ كانت فكرتنا في البداية بإستخدام خوارزمية التعلم الألة machine learning و تطبيقها على قاعدة بيانات لحسابات مستخدمي التويتر ثم محاولة تحسين دقتها بإستعمال خوارزمية التحسين الجسيمات PSO حيث تبين فعلا أن خوارزمية التحسين زادت في دقة النتائج.

أما بالنسبة لتحسين دقتها باستخدام خوارزمية EHO و بسبب ضيق الوقت، لم نتمكن من استنتاج نتيجة واضحة عنها.

---

**كلمات مفتاحية :** شبكة التواصل الاجتماعي، االتغريدات العشوائية, تويتر ,تحسين بسرب الجسيمات، تحسين رعي الفيلة, تعلم الآلة.

---

# Résumé

Les réseaux sociaux attirent des millions d'utilisateurs à travers le monde de défirent tranches d'ages et de catégorie, ce qui les qualifiés de devenir des plate-forme cibles pour les faux utilisateurs afin de diffuser une énorme quantité d'informations fausses et non pertinentes. Twitter, ne sort pas de cette problématique, et lui aussi exemple, est devenu l'une des plates-formes attirant les fraudeurs et les utilisateur mal-honnêtes qui diffusent des informations incorrectes aux utilisateurs via de fausses identités, ce qui a conduit à la propagation de contenus malveillants.

Ces derniers temps plusieurs approche on été proposé pour assurer la sécurité de ce réseau mais tous ces approches n'ont pas pus faire fin à ce défit, parmi ces approche on trouvent celle d'apprentissage automatique, approche bio-inspirés et bien-sure plusieurs autre approches. les approches d'apprentissage automatique ont données certains résultats, comme SVM.

Notre travail vise à étudie si on optimisant les caractéristiques par une approche bio-inspiré tel-que PSO, EHO est ce que cela peut améliorer les résultats obtenues. dans ce travail on va faire une comparaison selon deux stade : le premier au niveau de la methode SVM et sa combinaison avec PSO puis avec ECHO. au deuxième stade une comparaison entre SVM combiné avec PSO et SVM combinée avec EHO. les résultats ont montrés que SVM combinée avec PSO améliore la détection, mais pour les autres combinaison par manque de temps on a pas arrivée a en déduire une conclusion claire à leurs propos.

Dans ce travail, nous passons en revue les techniques utilisées pour détecter les faux utilateurs sur Twitter tout en essayant de développer une méthode de détection d'optimisation basée sur des algorithmes bio-inspirés tels que optimisation de l'essaim de particules (PSO) et optimisation de l'élevage d'éléphants (EHO).

Notre idée était initialement d'utiliser un algorithme d'apprentissage automatique et d'apprendre à l'appliquer à une base de données de comptes d'utilisateurs Twitter. Essayez ensuite d'améliorer sa précision en utilisant l'algorithme d'optimisation des particules PSO, où il a été constaté que l'algorithme d'optimisation augmentait la précision des résultats.


---


**Mots clés :** Réseau social, Twitter, Détection des Spammeurs, Optimisation de l'essaim de particules, Optimisation par l'élevage des éléphants, Apprentissage automatique.


---

# Abstract

Social networking sites are engaging millions of users around the world and these are becoming a target platform sites for fake users to spread huge amount of fake and irrelevant information. Twitter, for example, has become one of the most widely used platforms, increasing the possibility of spreading incorrect information to users via fake identities, which has led to the spread of malicious content in recent times. Contemporary Online Social.

In recent times, several approaches have been proposed to ensure the security of this network but not all of these approaches have been able to overcome this challenge. machine learning, bio-inspired approaches and of course many other approaches. machine learning approaches have given some results, such as SVM.

The aim of our work is to study whether optimizing features using a bio-inspired approach such as PSO or EHO can improve the results obtained. In the second stage, a comparison is made between SVM combined with PSO and SVM combined with EHO. The results show that SVM combined with PSO improves detection, but for the other combinations we haven't been able to draw any clear conclusions due to lack of time.

In this work, we review the techniques used to detect fake users on Twitter while attempting to develop an optimization detection method based on bio-inspired algorithms such as particle swarm optimization (PSO) and Elephant herding optimization (EHO).

Our initial idea was to use a machine-learning algorithm and ap to apply it to a database of Twitter user accounts. Then try to improve its accuracy using the PSO particle optimization algorithm, where it was found found that the optimization algorithm increased the accuracy of the results


---


**Keywords :** Social network, Spammer detection, Particle swarm optimization, Elephant herding optimization, Machine learning.


---

# Table des matières

# Liste des Figures

# Liste des Tables

# Liste des sigles et acronymes

**PSO** *Particle Swarm Optimization*

**EHO** *Elephant Herding Optimization*

**ML** *Machine Learning*

**SVM** *Support Vector Machine*

**FS** *Function selection*

**URL** *Uniform Resource Locator*

**CDF** *Cumulative distribution function*

# Introduction Générale

En raison de la popularité croissante des services de médias sociaux, les sites de réseaux sociaux tels que Facebook, MySpace et Twitter sont devenus l'un des principaux moyens pour les utilisateurs de suivre et de communiquer avec leurs amis en ligne

Malheureusement, la richesse d'informations existe dans ce type de média, ainsi que la facilité avec laquelle on peut atteindre de nombreux utilisateurs, ont également suscité l'intérêt de parties malveillantes, pour publier et échanger des contenus illégaux et suspects (images, vidéos, textes ...). En particulier, utilisateurs mal intentionnés dits "spammeurs" sont toujours à la recherche de moyens d'atteindre de nouvellesvictimes avec leurs messages non sollicités. C'est ce que montre uneétude de marché sur la perception du spam par les utilisateurs sur les réseaux sociaux.

Détecter les spammeurs est un réel défi pour maintenir un haut niveau de performance dans les applications mises en œuvre dans les réseaux sociaux, tels que Twitter (comme exemple) qui est devenu l'une des plateformes les plus utilisées et qui autorise donc une quantité anormal de spam où les faux utilisateurs (les spammeurs) envoient des tweets indésirables aux utilisateurs ce qui non seulement perturbe les utilisateurs légitimes mais gaspille également des ressources. Dans ce travail l'idée est de, nous explorons des méthodes et stratégies pour classer un utilisateur comme spammeur ou non-spammeur, par l'identification de certain nombre de caractéristiques liées au contenu des tweets.

Un grand nombre de problèmes peuvent, en effet, être décrits sous la forme de problèmes d'optimisation. C'est le cas, par exemple, pour les problèmes de la découverte des spammeurs dans les réseaux sociaux, où de nombreux algorithmes inspirés de la nature sont proposés au cours des dernières décennies, parmi ceux-ci, les algorithmes bio-inspirés tels que les algorithmes révolutionnaires, ces algorithmes sont caractérisés par des possibilités de recherche globale,d'une capacité de convergence rapide, leur robustesse et leur capacité à trouver de bonnes solutions dans un vaste espace de recherche.

Parmi ces algorithmes méta-heuristiques, le paradigme de l'optimisation de l'essaimage de **(PSO)**, optimisation du troupeau (clan) d'éléphants **(EHO)**, qui provient du comportement des animaux sociaux, pour résoudre des tâches d'optimisation globale, qui s'inspire du comportement de groupe (intelligence en essaims). L'objectif de notre étude et d'élaborer un algorithme de détection des spammeur dans un réseau social (twitter), basé sur les deux techniques méta-heuristique **PSO** et **EHO**, en suite on présente une comparaison entre les deux algorithmes

par-rapport les résultats.

Nous organiserons notre mémoire comme suit :

Le chapitre I : Généralité nous exposons les notions, les définitions de base des différentes concepts utilisées dans notre projet.

Le chapitre II : Dans ce chapitre, nous expliquons le fonctionnement des algorithmes d'optimisation pour PSO et EHO qui font l'objet de notre travail.

Le chapitre III : un état de l'art présente une synthèse des travaux relatifs au sujet et les contributions des chercheurs concernant les techniques qu'on va les utiliser ultérieurement dans la partie pratique.

Le chapitre IV : Implémentation et expérimentation des approches appliquées pour résoudre ce problème avec une discussion des résultats obtenus.

Une conclusion termine notre travail.

# Chapitre 1

# Notions Préliminaires

Les réseaux sociaux sont considère comme un moyen populaire pour les utilisateurs de s'interagir entre eux, les interactions des utilisateurs avec les sites sociaux ont parfois des résultats négatifs qui influe négativement aussi sur la fiabilité du système. Les utilisateurs non sollicités ont transformé ces sites de médias sociaux en cibles pour la transmission des informations fausses et dangereuses. Dans ce premier chapitre, nous présentons les définitions, les notions et les concepts de base utilisés dans notre travail.

## 1.1 Problèmes de sécurité dans les réseaux sociaux

La demande accrue de sites sociaux permet aux utilisateurs de collecter une quantité abondante d'informations et de données sur les utilisateurs. Les énormes volumes de données disponibles sur ces sites attirent également l'attention de faux utilisateurs, Twitter, par exemple, est devenu l'une des plateformes les plus utilisées de tous les temps et autorise donc une quantité déraisonnable de contenu non sollicitée diffusés par les utilisateurs indésirables dite (les spammeurs). Récemment, la détection des spammeurs et l'identification des faux utilisateurs sur Twitter sont devenues un domaine de recherche commun dans les réseaux sociaux [1].

### 1.1.1 Types de spammeurs

Tous d'abord un spammeur c'est un utilisateurs malveillants (indésirable) qui a comme but la contamination des informations présentées par d'autres utilisateurs légitimes et constituent à leur tour un danger pour la sécurité et la confidentialité des réseaux sociaux. Les différents catégories des spammeurs [1] :

1. les Faux utilisateurs (fake users) [1] : ce sont des utilisateurs qui se font passer pour de vrais profils d'utilisateurs afin d'envoyer du contenu de spam aux amis de l'utilisateur et aux autres utilisateurs du réseau.

2. les Hameçonneurs (Phishers) [1] : utilisateurs qui agissent comme des utilisateurs réguliers et obtiennent des informations personnelles d'autres utilisateurs réels.

3. Promoteurs (promoters) [1] : personnes qui envoient des publicités nuisibles ou d'autres liens promotionnels à des tiers afin d'obtenir des informations personnelles.

### 1.1.2 Principaux dangers des spammeurs sur les réseaux sociaux

Sites de réseaux sociaux en ligne (online social network) tel que twitter facebook…. sont vulnérables aux problèmes de sécurité et de confidentialité vu que la quantité d'informations des utilisateurs que ces sites traitent quotidiennement .Les utilisateurs des réseaux sociaux font l'objet de diverses attaques :

1. - Le spam (pourriel) c'est un mail indésirable, envoyé massivement par le spammeur aux utilisateurs des sites des réseaux sociaux [2].

2. - Le hameçonnage c'est de la cybercriminalité (pishing), Ce type d'attaque vise à inciter le destinataire d'un email d'apparence légitime à fournir ses informations d'identification bancaires ou financières afin de voler de l'argent.

3. - Les spammeurs envoi des programmes malveillants sur les réseaux sociaux (virus) pour pirater des information ou infecter le système [2].

4. - Attaque Sybil (fausse) - Les attaquants reçoivent plusieurs fausses identités pour prétendre être réels au sein du système et nuire à la réputation des utilisateurs honnêtes du réseaux [3].

5. - Les bots sociaux prétendent être des utilisateurs humains sur les réseaux sociaux. Ils ne donnent aucune idée qu'ils sont des machines. sont classés comme "faux comptes" [1].

6. - Attaques d'usurpation d'identité et de clonage : les attaquants créent le profil d'un utilisateur qui existe déjà sur le même réseau ou sur d'autres réseaux pour tromper des amis [1].


Fig. 1.1 : Dangers des spammeurs sur les réseaux sociaux.

## 1.2 Twitter comme plateforme réseaux sociaux

Twitter est un site de service de réseau social lancé le 21 mars 2006 compte 500 millions d'utilisateurs actifs [1] à ce jour qui partagent information.le logo de Twitter est un image d' un oiseau gazouillant sous le nom « Twitter ». Les utilisateurs peuvent y accéder pour échanger fréquemment des informations appelées « tweets », qui sont des messages que n'importe qui peut envoyer ou lire d'une longueur maximale de 140 caractères. Ces tweets sont publics par défaut, Toute personne qui vous suit sur Twitter peut le voir. Les utilisateurs partagent des Tweets contenant des actualités, des opinions, des photos, des vidéos, des liens et des messages. Les termes standard utilisés sur Twitter et liés à notre travail incluent :

. Le nom d'utilisateur Twitter [4] : Un identifiant Twitter est simplement votre nom d'utilisateur sur Twitter. Il vous sera demandé de créer ce nom lorsque vous vous inscrirez pour utiliser Twitter. Ce nom est le nom que vous utilisez pour vous connecter à votre compte Twitter et c'est le nom par lequel les autres utilisateurs de Twitter vous reconnaîtront. Cela ne fait pas de mal de prendre un peu de temps et de réfléchir un peu. Un bon surnom est celui qui vous identifie facilement, mais qui est un peu créatif.

. Tweet[4] :est un message Twitter de 140 caractères maximum.

. Followers[4] : Il existe de nombreux utilisateurs de Twitter dans le monde, mais cela ne signifie pas que tout le monde peut voir vos publications. Donc les personnes qui ont choisi de vous suivre et de recevoir des mises à jour sur vos publications appelons Les abonnés , sachant que ces mises à jour apparaîtront dans votre flux Twitter .

. Retweet [4] : recevez un tweet et partagez-le en tant que nouveau tweet avec ces abonnés (followers ) du réseau social Twitter.

. Hashtag [4] : Mot-clé précédé du symbole # utilisé par les internautes dans leurs publications sur les réseaux sociaux. Ils permettent à d'autres utilisateurs d'accéder au contenu contenant le mot-clé ci-dessus sans nécessairement être des "amis" ou des "abonnés" de la personne qui l'utilise.

. Mention[4] : les tweets peuvent inclure des réponses et des mentions d'autres utilisateurs en faisant précéder leurs noms d'utilisateur du signe.

. Listes [4] : Twitter fournit un mécanisme pour répertorier les utilisateurs que vous suivez dans des groupes.

. Message direct [4] :Également connu sous le nom de DM, le système de messagerie directe de Twitter pour la communication privée entre les utilisateurs.

### 1.2.1 Menaces sur Twitter

1. Tweets spammés[1] : Twitter permet à ses utilisateurs de publier des tweets de 140 caractères maximum, mais quelle que soit la limite de caractères, les cybercriminels ont trouvé un moyen d'utiliser cette limitation à leur avantage en créant des tweets courts mais convaincants avec des liens vers des promotions pour des bons gratuits ou des publications d'offres d'emploi ou d'autres promotions.

2. Téléchargements de logiciels malveillants [1] : Twitter a été utilisé par les cybercriminels pour diffuser des publications contenant des liens vers des pages de téléchargement de logiciels malveillants. Les applications FAKEAV et backdoor[13] sont des exemples de vers Twitter qui expédié des messages directs et même des logiciels malveillants affectant à la fois les systèmes d'exploitation Windows et Mac. Le malware de réseaux sociaux le plus terni est KOOBFACE [1], qui ciblait à la fois Twitter et Facebook.

3. Robots Twitter [1] : Les cybercriminels ont tendance à utiliser Twitter pour gérer et contrôler les botnets. Ces botnets contrôlent les comptes des utilisateurs et constituent une menace pour leur sécurité et leur vie privée.

### 1.2.2 Distinction des spammeurs et non spammeurs sur twitter

Les fonctionnalités sur la base des quelles les profils spam et non spam sont différenciés sellons l'utilisateur,le contenu,utilisateur avec le contenu ainsi que d'autres aspects parmi les quelles on peut citer distance ou la connectivité graphique : [5]

1. **Caractéristiques basées sur l'utilisateur :** Qui incluent des caractéristiques démographiques telles que les détails du profil, le nombre d'abonnés, le ratio abonnés/abonnés, la réputation, l'âge du compte, la moyenne. temps entre les tweets affichant le comportement du temps, les heures d'inactivité, la fréquence des tweets, etc.

2. **Caractéristiques basées sur le contenu (tweets) :** Qui incluent le nombre de hashtags , le nombre d'URL dans les tweets, les mentions @, les retweets, les mots de spam, les liens HTTP, les sujets tendance, les tweets en double, etc.

3. **Caractéristiques basées sur l'utilisateur et le contenu à la fois.**

4. **Autre caractéristiques :** comme méthode de clustering de Markov, taux d'URL, taux d'interaction, relations sociales, activités sociales, fonctionnalités basées sur les graphes, fonctionnalités basées sur les voisins, fonctionnalités basées sur l'automatisation

Rôle des attributs mentionnées ci-dessus pour la détection de profil de spam conformément à la politique de Twitter [1] :

- **Nombre d'abonnés(followers) :**les spammeurs ont moins d'abonnés.

- **Nombre de suivis(followings)** : Les spammeurs suivent généralement un grand nombre d'utilisateurs.
- **abonnés/suivis Rapport :** ce rapport est inférieur à 1 pour les spammeurs.
- **La réputation :** est définie comme le rapport des abonnés à la somme des abonnés et suivis. Les spammeurs ont une réputation
- **Âge du compte :** déterminé en fonction de la date actuelle et de la date de création du compte. Les spammeurs ont généralement de nouveaux comptes, donc cette fonctionnalité a moins de valeur pour les spammeurs.
- **Temps moyen entre les publications :** afin d'attirer l'attention des autres,les spammeurs publient plus de tweets en peu de temps
- **Comportement à l'heure de publication :** Les spammeurs ont tendance à publier selon un horaire fixe, cela peut être tôt le matin ou tard le soir lorsque les vrais utilisateurs n'utilisent pas les sites de réseaux sociaux
- **l'heures d'inactivité :** Les spammeurs continuent d'envoyer des messages afin qu'ils aient moins d'heures d'inactivité.
- **Nombre d'URL :** Les tweets des spammeurs se composent d'un grand nombre d'URL des sites malveillants.

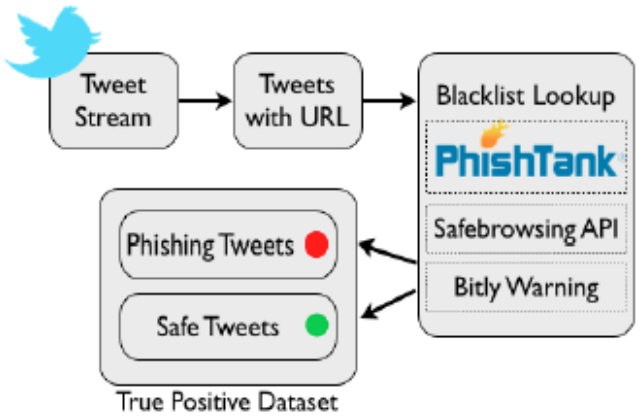


Fig. 1.2 : Détection de spammeur basée sur l'URL.
[6]

- **Fréquence des tweets :** les spammeurs publient fréquemment des tweets à des moments impairs pour attirer l'attention des autres utilisateurs.
- **Nombre de hashtages :** les spammeurs tweetent plusieurs mises à jour sans rapport avec les sujets les plus mentionnés sur Twitter en utilisant # pour inciter les utilisateurs légitimes à lire leurs tweets.

- **mentions :** les spammeurs utilisent au maximum les @noms d'utilisateurs inconnus dans leurs tweets afin d'éviter d'être détectés.
- **Retweets :** les retweets sont des réponses aux tweets avec le symbole @RT, et les spammeurs utilisent @RT autant que possible dans leurs tweets.
- **Mots de spam :** les tweets des spammeurs sont principalement constitués de mots indésirables.
- **Liens HTTP :** si les tweets contiennent un nombre élevé de www ou http ://, alors ils sont postés par des spammeurs.
- **Tweets en double :** les spammeurs ont tendance à publier des tweets en double avec différents noms d'utilisateur dans les tweets.

## 1.3 Fonctionnalités basées sur l'utilisateur

La détection de faux comptes est devenue un sujet brûlant car de nombreux réseaux sociaux en ligne (OSN) rencontrent des problèmes causés par l'augmentation des activités sociales en ligne contraires à l'éthique.

Dans Ce travail, nous introduisons la fonction de distribution cumulative (CDF) pour caractériser les spammeurs, comme le montre la figure (1.3).[7]

(1.3(a)) traite et analyse le nombre des personnes suivies pour chaque utilisateur. Les normes dit, le spammeur essaient de suivre une multitude d'utilisateurs légitimes afin d'être suivis en retour. Malgré que, cela ne répond pas la plupart du temps, comme le montre (1.3(b)).

Ce type de comportement fait que la fraction de "followees" par "followers" est très importante par rapport aux légitimes utilisateur (non spammeur), comme le montre la fig (1.3(c)). L'analyse de l'age (nombre de jours de création) voir Fig(1.3(d)). Indique que les spammeurs doivent créer de nouveaux comptes fréquemment. Cela peut construire un mécanisme anti-spammeur qui, à terme, détecterait et nettoierait automatiquement les comptes des spammeurs.

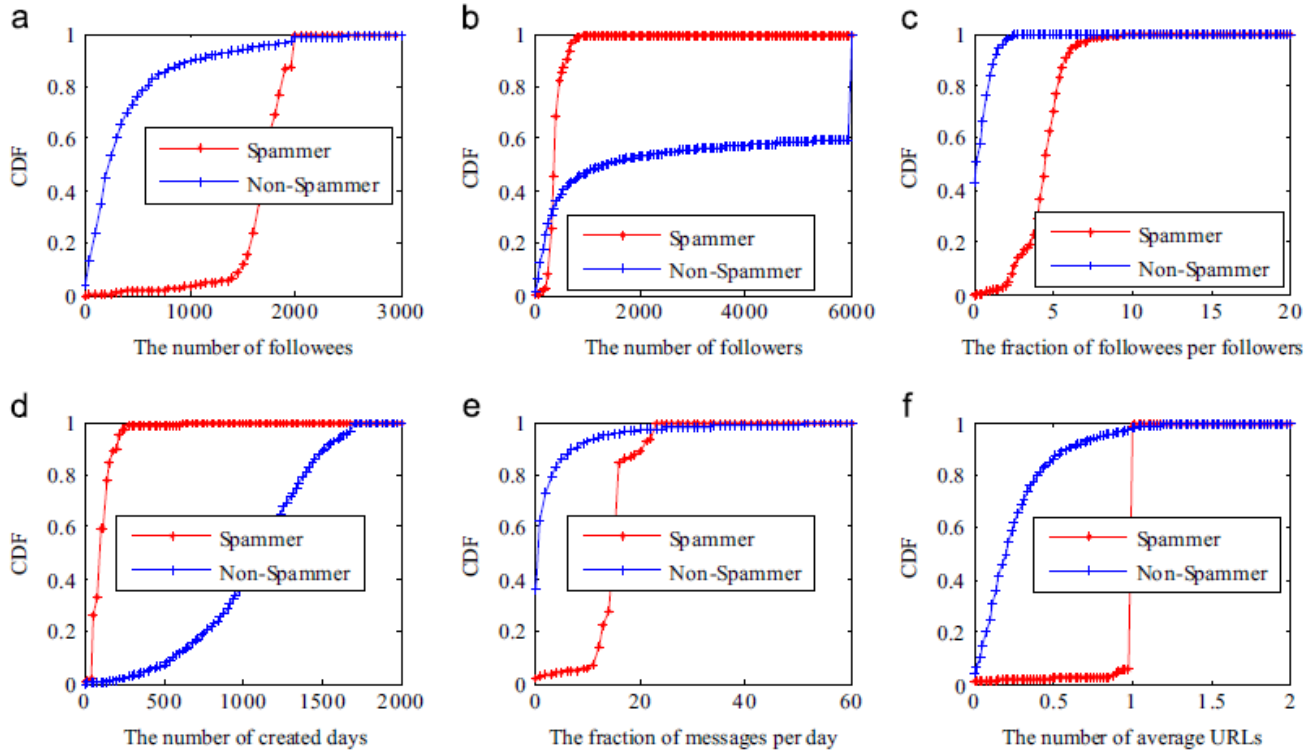


Fig. 1.3 : Fonction de distribution cumulative de la caractéristique basée sur l'utilisateur. [7]

## 1.4 Apprentissage automatique pour la détection des spammeurs sur TWITTER

L'apprentissage automatique (ML) est une branche de l'intelligence artificielle qui permet aux machines d'utiliser les données disponibles pour prédire les actions, les résultats et les futurs développements . Grâce à ce procédé, les machines apprennent sans automatisation externe. C'est l'apprentissage logique d'algorithmes et de modèles statistiques que les machines utilisent pour effectuer des tâches spécifiques basées sur des modèles sans l'utilisation de commandes externes.

### 1.4.1 Modeles de l'apprentissage automatique

une machine est formée pour apprendre de l'expérience E en ce qui concerne l'ensemble des tâches T et la performance P si sa performance aux tâches T progresse avec l'expérience E. Parmi les techniques d'apprentissage automatique les plus utilisé pour la détection des spammeurs dans twitter : Naïve Bayes, Support Vector Machine (SVM) et Random Forest.[7]

**Machine à vecteurs de support (SVM) [8] :** cette technique d'apprentissage automatique est utilisée pour résoudre des problèmes de classification et de régression. Son but est de trouver un hyperplan dans un espace à n dimensions (n étant le nombre de variables d'entrée) qui catégorise les points de données.SVM fait partie des travails de la détection de visage, la clas-

sification de texte, la classification d'images, ... etc.

### 1.4.2 Modèle de détection de spammeurs basé sur un SVM

illustre le concept de base du modèle de détection des spammeurs proposé. Dans cette solution, les données d'apprentissage sont converties en une série de vecteurs de caractéristiques qui consistent en un ensemble de valeurs pour les attributs [7]. Ces vecteurs constituent l'entrée de l' algorithme d'apprentissage machine supervisé. Après l'apprentissage, un modèle de classification est appliqué pour déterminer si l'utilisateur appartient à la catégorie des utilisateurs normaux ou à celle des spammeurs. Les spammeurs et les non spammeurs ayant des comportements sociaux différents, l'analyse des caractéristiques du contenu et du comportement des utilisateurs permet de distinguer les comportements anormaux des comportements légitimes. Dans cet travail, nous avons défini les caractéristiques énumérées comme suit : le nombre de followees, le nombre de followers, le nombre de messages, le nombre d'amis qui se suivent, le nombre de favoris, le nombre de jours de création, la fraction de followees par followers, la fraction de messages originaux, le nombre de messages par jour, le nombre moyen de reposts, le nombre moyen de commentaires, le nombre moyen de likes, le nombre moyen d'URL, le nombre moyen d'images, le nombre moyen de hashtags, le nombre moyen d'utilisateurs mentionnés, la fraction de messages contenant des URL, la fraction de messages contenant des images. [8]

## 1.5 Conclusion

Dans ce chapitre on a présenter des notions de base sur les réseau sociaux, en particulier twitter, les spammeurs (utilisateurs indésirables) et leurs dangers, ainsi que la détection des spams sur la plat-forme twitter et ces principes caractéristiques

# Chapitre 2

# Méthodes d'optimisation bio-inspiré

## 2.1 Présentation

Les défis actuels des systèmes de détection des spammeurs sont directement liés à la faible précision de la classification des spammeurs et à la grande dimensionnalité du processus de sélection des fonctionnalités. Cependant, la sélection des fonctionnalités (SF) est une approche d'optimisation globale de l'apprentissage automatique qui réduit la redondance des données et produit des ensembles de résultats précis et acceptables. La nature trouve toujours les meilleures solutions pour résoudre les problèmes et maintenir le parfait équilibre entre ses composants. Les systèmes de bioinspiration, comme leur nom l'indique, sont des systèmes métaheuristiques qui imitent des phénomènes observés dans la nature pour résoudre de nombreux problèmes dans divers domaines. Cet article présente deux algorithmes d'optimisation inspirés de la biologie, tels que l'optimisation par essaim de particules (PSO). et l'optimisation de l'élevage d'éléphants EHO sont améliorés pour réduire la dimensionnalité des fonctionnalités et améliorer la précision de la classification des spammeurs.

Les méthodes bio-inspirés peuvent être réparties en deux grandes classes selon la source d'inspiration de la méthode bio-inspiré,est représenté sur la Figure (2.1).

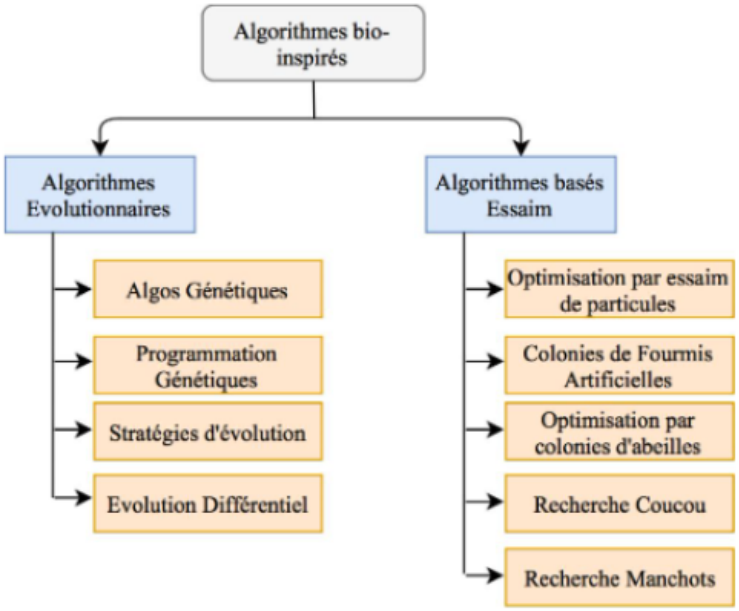


Fig. 2.1 : Classification d'algorithmes bio-inspirés.
[9]

## 2.2 Optimisation par Essaim de Particules (PSO)

Une méta-heuristique est une technique d'optimisation qui permet d'obtenir une approximation de la solution optimale en un temps raisonnable. Ils sont destinés à résoudre de nombreux problèmes dans différents domaines sans changer les principes algorithmiques de base de la méthode. Une attention particulière est portée à la technique d'optimisation approchée PSO. La méthode trouve son origine dans les observations faites par Reynolds et Heppner-Grenander dans des simulations informatiques de vol groupés d'oiseaux fig(2.2)[1]. Il est basé sur des « interactions sociales » entre « agents », appelées « particules », visant à atteindre des objectifs spécifiques dans un espace de recherche partagé. Chaque particule a une capacité spécifique à stocker et à traiter des informations. Cette métaheuristique d'optimisation stochastique a été proposée en 1995 par le psychologue social James Kennedy et l'ingénieur électricien Russell Eberhart.[10, 11, 12]

Fig. 2.2 : Vol groupé d'oiseaux.

### 2.2.1 Méthode PSO

Dans PSO, le comportement social est modélisé par des formules mathématiques qui permettent de contrôler le processus de migration des particules .[10] Le mouvement des particules est influencé par trois facteurs : inertiel, cognitif et social. Chacun de ces composants reflète une partie de l'équation :

- **Composante d'inertie :** les particules ont tendance à suivre la direction actuelle du mouvement.
- **Facteur cognitif :** les particules ont tendance à se déplacer vers les meilleurs endroits où elles sont déjà passées.
- **Facteur social :** les particules ont tendance à se déplacer vers des positions optimales que les particules voisines peuvent atteindre.

[1] https ://www.lapresse.ca/actualites/sciences/201502/02/01-4840716-les-oiseaux-migrateurs-se-relaient-en-tete-pour-moins-se-fatiguer.php

Le principe du mouvement des essaims de particules est résumé à la figure (2.3)[2].

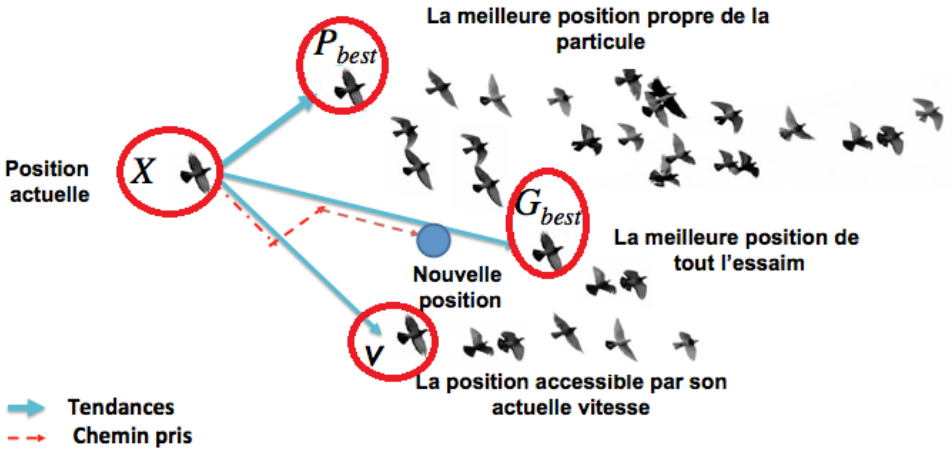


Fig. 2.3 : Déplacement d'une particule..

### 2.2.2 Principes de base

Dans l'espace de recherche de dimension ***d*** , une particule de l'essaim ***i*** est représentée par son vecteur position $\overrightarrow{X_i}$ et par son vecteur vitesse $\overrightarrow{V_i}$ , est formulé comme suit [13] :

$$\overrightarrow{X_i} = (X_1, X_2, ..., X_d)$$
$$\overrightarrow{V_i} = (V_1, V_2, ..., V_d)$$

L'appréciation de la qualité de sa position est arrêtée par la valeur de la fonction "objectif" en ce point. Il est indispensable que cette particule puisse mémoriser la meilleure position par laquelle elle est déjà passée, formulée comme suit :

$$\overrightarrow{P}best_i = (Pbest_1, Pbest_2, ..., Pbest_D)$$

La meilleure position atteinte par les particules de l'essaim est formulée comme suit :

$$\overrightarrow{G}best_i = (Gbest_1, Gbest_2, ..., Gbest_D)$$

Le terme *Gbest* (Global Best) signifie que toutes les particules de l'essaim proviennent de la particule ***i***. Au début de l'algorithme, les particules de l'essaim sont initialisées de manière aléatoire ou périodique dans l'espace de recherche. Après cela, chaque itération déplace les particules et fusionne les trois composants ci-dessus.

[2]https ://www.researchgate.net/figure/Deplacement-dune-particule$_{fig1_3}$15851237

$wv^t_{i.j}$ : représente la composante d'inertie du déplacement, où le paramètre ***w*** gère l'influence de la direction de déplacement sur le déplacement futur.

$c_1 r^t_{1i.j}[pbest^t_{i.j} - x^t_{i.j}]$ : représente la composante cognitive du déplacement, où le paramètre C1 gère le comportement cognitif de la particule

$c_2 r^t_{2i.j}[gbest^t_{i.j} - x^t_{i.j}]$ : représente la composante sociale du déplacement où le paramètre C2 gère l'aptitude sociale de la particule.

En fusionnant ces trois composantes, nous pouvons d'abord trouver la vitesse, ce qui nous permet de calculer la position de la particule. Ces deux variables sont à leur tour déterminées à l'aide des équations (2.1) et (2.2) comme suit :

$$v^{t+1}_{i.j} = wv^t_{i.j} + c_1 r^t_{1i.j}[pbest^t_{i.j} - x^t_{i.j}] + c_2 r^t_{2i.j}[gbest^t_{i.j} - x^t_{i.j}] \qquad \mathbf{j} \in [1, 2, ..., d] \tag{2.1}$$

$$x^{t+1}_{i.j} = x^t_{i.j} + v^{t+1}_{i.j} \qquad \mathbf{j} \in [1, 2, ..., d] \tag{2.2}$$

- w est une constante, appelée coefficient/pourcentage d'inertie.
- $c_1$, $c_2$ sont deux constantes, appelées coefficients/pourcentage d'accélération.
- $r_1$ , $r_2$ sont deux nombres aléatoires tirés uniformément dans [0, 1].

et ce à chaque itération **t** et pour chaque dimension **j**. A la fin du déplacement des particules dans une itération donnée, les nouvelles positions sont évaluées et les deux vecteurs $Pbest_i$ et $gbest_i$ sont réindexés.

### 2.2.3 Algorithme PSO

L'algorithme de base de la méthode PSO proposée par [KEN 95] commence par initialiser aléatoirement les particules dans l'espace de recherche en leur attribuant des positions et des vitesses initiales.

A chaque itération de l'algorithme, les particules sont déplacées selon les équations (2.1) et (2.2) et la fonction objectif des particules (fitness) est calculée de sorte que les meilleures positions de tous les $Pbest_i$ puissent être calculées.

$Pbest_i$ et $Gbest_i$ sont mis à jour à chaque itération selon l'algorithme décrit dans (Algorithme 1).

Le processus est répété jusqu'à satisfaction du critère d'arrêt.

**Début**
  **Pour chaque particule**:
    On initialise sa position
    On initialise sa meilleure position $p$ connue comme étant sa position initiale
    **Si** $f(p) < f(g)$
      On met à jour la meilleure position de l'essaim
    **Fin si**
    On initialise la vitesse de la particule
  **Fin pour**
  **Tant que** le critère d'arrêt n'est pas vérifié (on n'a pas atteint l'itération maximale):
    **Pour chaque particule** $i$ :
      On tire aléatoire $r1$ et $r2$
      On met à jour la vitesse de la particule selon l'équation (2.1)
      On met à jour la position $xi$ selon l'équation (2.2)
      **Si** $f(xi) < f(pi)$
        On met à jour la meilleure position de la particule
      **Fin si**
      **Si** $f(pi) < f(g)$
        On met à jour la meilleure position de l'essaim
      **Fin si**
    **Fin pour**
  **Fin tant que**
**Fin**

**Algorithme 1: Optimisation par Essaim de Particules (PSO).**

## 2.3 Optimisation l'élevage d'éléphants (EHO)

Fin 2015, Wang a proposé Optimisation de L'élevage d'éléphants (EHO), une méthode de recherche métahoristique intelligente basée sur un essaim, pour résoudre les problèmes d'optimisation. L'algorithme est basé sur la modélisation du comportement des troupeaux d'éléphants sauvages[14]. Le comportement du troupeau peut être visualisé comme suit :

- Les troupeaux d'éléphants sont constitués de nombreux sous-groupes appelés clans, dont chacun se compose d'un certain nombre d'éléphants femelles et d'éléphants mâles [15].Voir Figure 2.4.
- La direction du clan ou du groupe dans ses déplacements est sous la supervision de la mère (une éléphante adulte) voir Fig (2.4). [15]
- Les éléphants mâles quittent le clan auquel ils appartiennent après avoir atteint l'âge adulte.

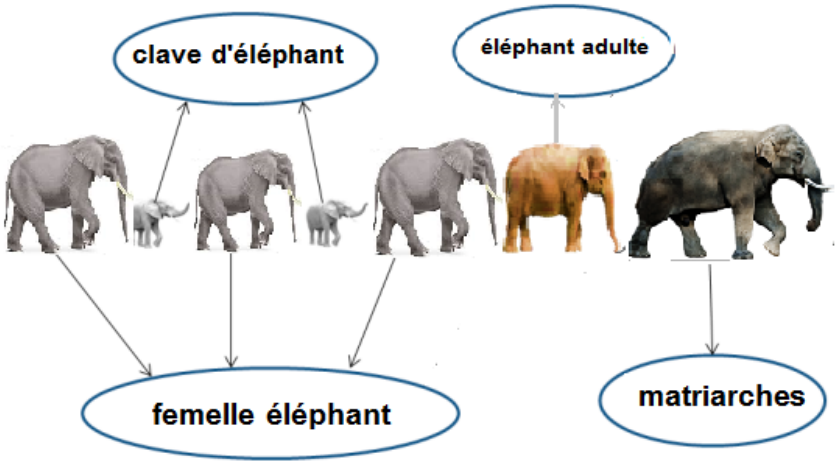


Fig. 2.4 : Population d'éléphants.
[16]

### 2.3.1 Description de la méthode EHO

Les éléphants sont les animaux les plus grands et les plus sociaux. Il existe deux types d'éléphants traditionnellement reconnus,les éléphants d'Afrique et les éléphants d'Asie. Caractérisé par un long trompe, il a de multiples fonctions telles que tenir des objets, soulever de l'eau et respirer. Les éléphants ont des comportements sociaux complexes avec les femelles et les petits. Les éléphants vivent en groupes multi-claniques dirigés par des mères plus âgées. Un clan se compose d'une femelle et de sa progéniture, ou de plusieurs femelles. Ils préfèrent vivre en troupeaux familiaux, mais les éléphants mâles ont tendance à vivre isolés et à quitter leurs troupeaux familiaux lorsqu'ils atteignent l'âge adulte. Les éléphants mâles vivent à l'écart du troupeau, mais des vibrations à basse fréquence leur permettent de rester en contact avec les éléphants de leur clan [16, 14, 15].

L'algorithme EHO analyse le comportement de pâturage des éléphants en deux mécanismes[14] :

1. Mécanisme de mise à jour du clan :

   Les éléphantes femelles de chaque clan vivent sous la direction de leurs matriarches (femelles adultes). Le statut des autres éléphants au sein du clan **ci** est influencé par le statut de la matriarches, de cette manière le statut de l'éléphant **j** dans le clan **ci** est mis à jour,en utilisant l'éq. (2.3) .

$$x_{new,ci,j} = x_{ci,j} + \alpha * (x_{best,ci} - x_{ci,j}) * r \tag{2.3}$$

   où $x_{ci.j}$ dénote l'ancienne position et $x_{new,ci,j}$ dénote la nouvelle position mise à jour pour l'éléphant ***j*** dans le clan ***ci***, respectivement [15]. $\alpha \in [0,1]$ est un facteur d'échelle qui détermine l'influence de la matriarche $c_i$ sur $x_{ci.j}$ , $x_{best,ci}$ représente la matriarche $c_i$, qui est l'individu le plus apte à devenir éléphant dans le clan $\boldsymbol{ci}.r \in [0,1]$. On utilise ici une distribution uniforme. L'éléphant le plus apte dans chaque clan ne peut pas être mis à jour

par l'équation (2.3) , c'est-à-dire que $x_{ci.j}$=$x_{best,ci}$. Pour l'éléphant le plus apte, il peut être mis à jour par l'équation (2.4). :

$$x_{new,ci,j}=\beta * x_{center,ci} \tag{2.4}$$

où $x_{center,ci}$ représente le centre du clan ***ci*** généré à partir des informations obtenues par les éléphants du clan ***ci*** [16]. $\beta \in [0,1]$ est un opérateur représentant l'influence de $x_{center,ci}$ sur $x_{new,ci,j}$ . $x_{center,ci}$ de dimension ***d*** peut être calculé par l'équation (2.5).

$$x_{center,ci,d} = \frac{1}{n_c i} \sum_{j=1}^{n_c i} x_{ci,j,d} \tag{2.5}$$

où ***d*** est la dimension de 1 au nombre total de dimensions $(1 \leq d \leq D)$. $n_{ci}$ représente la population du clan ***ci*** et $x_{ci,j,d}$ représente le ***d*** éléphant $x_{ci,j}$ [14]. $x_{center,ci}$ représente le centre du clan ***ci*** et peut être calculé à l'aide de l'équation (2.5).

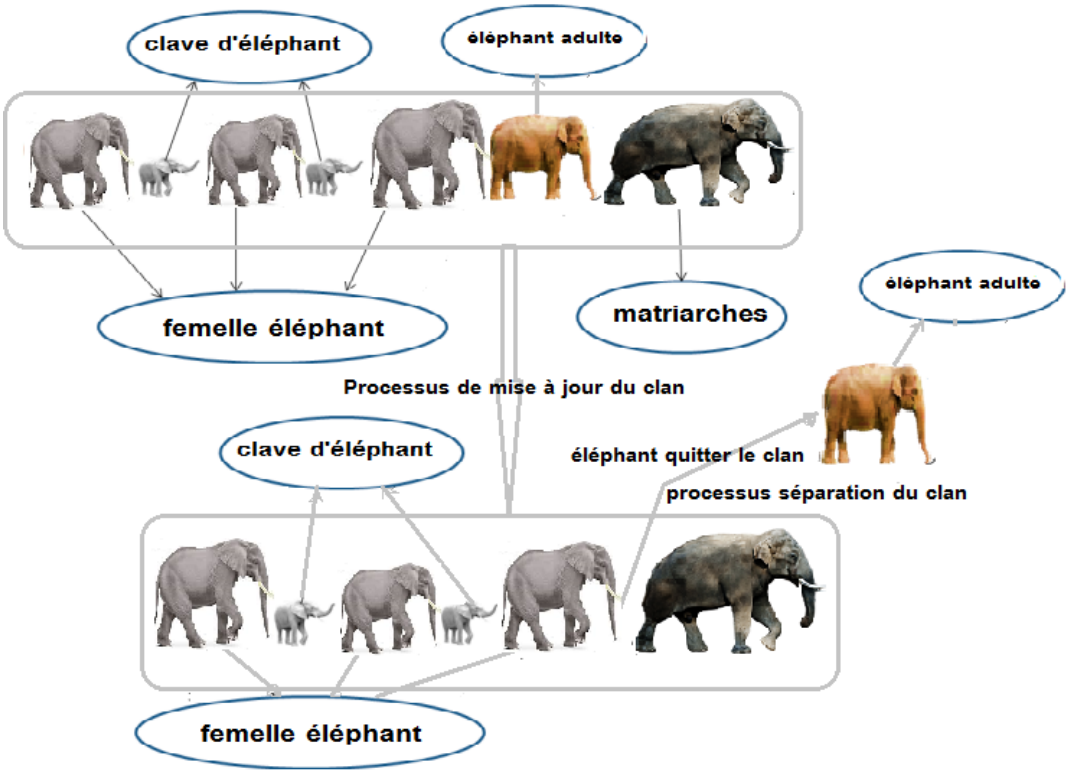


Fig. 2.5 : Mise à jour du clan.
[16]

2. Mécanisme de séparation :

Lorsque les éléphants mâles grandissent, ils quittent leur famille et vivent seuls voir fig (2.5). Ce processus de séparation peut être conçu pour séparer les opérateurs lors de la

résolution d'un problème d'optimisation. [14] Pour améliorer encore la capacité de recherche de l'algorithme EHO, nous supposons que les individus d'éléphants dans la pire condition physique appliquent un facteur de ségrégation à chaque génération, comme le montre l'équation (2.6).

$$x_{worst,ci} = x_{min} + (x_{max} - x_{min} + 1) * r \tag{2.6}$$

Ici, $x_{wost,ci}$ désigne le pire éléphant mâle du clan ***ci***. $x_{max}$ et $x_{min}$ indiquent les limites inférieure et supérieure de la position de l'éléphant. $rand \in [0, 1]$ est une sorte de distribution stochastique. [15, 16]

### 2.3.2 Algorithme EHO

Sur la base des règles présentés dans la partie précédente, les différentes étapes de la métaheuristique EHO sont résumées dans l'algorithme $x_{new,ci,j}$ et $x_{best,ci}$ sont mis à jour à chaque itération selon l'algorithme décrit dans (Algorithme 2).

```
Begin
  Initialisation. Définissez les itérations d'initialisation G = 1
  Initialiser la population P au aléatoire
  Définir la génération maximale MaxGen
  Tant que le critère d'arrêt n'est pas satisfait faire
    Trier la population en fonction de l'aptitude des individus.
    Pour tous les clans ci do
      Pour l'éléphant j dans le clan ci do
        Générer xnew, ci, j et mettre à jour xci, j          selon l'équation (2.3)
        Si xci, j = xbest, ci alors
          Générer xnew, ci, j et mettre à jour xci, j        selon l'équation (2.4)
        Fin Si
      Fin Pour
    Fin Pour
    Pour tous les clans ci do
      Remplacer le pire ci individuel                        selon l'équation (2.6)

    Fin Pour
    Évaluez chaque éléphant en fonction de sa position.
    T = T + 1.
  Fin Tant que
Fin
```

**Algorithme 2 : Optimisation L'élevage d'éléphants(EHO).**

## 2.4 Conclusion

Dans ce chapitre nous avons présenté la méthode moderne d'optimisation par algorithmes basée sur les techniques métaheuristiques **PSO** et **EHO**. En les présentant et en expliquant et discutant l'origine et le principe de cette méthode.

Nous avons également expliqué le principe et les bases du fonctionnement de ces algorithmes PSO et EHO.

# Chapitre 3

# Etat de L'art

## 3.1 Présentation

Les méthodes de détection des spammeur ou les utilisateurs indésirables en général, sont de plus en plus évolué. Nous présenterons dans ce chapitre un aperçu sur quelques travaux réalisés auparavant qui ont une relation à notre sujet, proposant des méthodes de sélection de caractéristiques (attribus)/ (Features Sélection) basées sur des algorithmes bio-inspirés dérivés de l'intelligence en essaim. Ces approches peuvent être considérées comme des tendances majeures de la recherche dans le domaine de filtrage des spammeurs.

## 3.2 Optimisation des méthodes de détection

L'objectif principal était d'améliorer le temps de calcul et maximiser le taux de précision des algorithmes d'apprentissage automatique Plusieurs chercheurs ont adopté la sélection d'attributs basée sur PSO pour diverses applications,dans la suite on va présenter brièvement quelques algorithmes.

**travaux réalisés basé sur le contenu :**

Par exemple, Liu et al. Dans [17] ont proposé une nouvelle méthode d'apprentissage (PSO-LM ) pour processus de propagation des réseaux de neurones (PNN) basés sur l'optimisation des essaims de particules (PSO) et les fonctions de mélange gaussiennes. Les résultats des expériences ont montré que l'application (PSO-LM) sur l'ensemble de données spambase a atteint une précision de 90,5 réseaux de neurones à rétropropagation (BPNN) et apprentissage basé sur l'expansion des fonctions de base méthode (BFE-LM).

Ye et al.[18] ils ont utilisé BPSO, un algorithme PSO binaire amélioré en combinaison avec un mécanisme immunitaire pour résoudre le problème d'optimisation transformé.

Awad et Foqaha. dans [19] ont proposé un algorithme hybride de réseau de neurones rbf et de particules optimisation en essaim (HC-RBFPSO) pour la classification des spams. Ils ont utilisé un essaim de particules algorithme d'optimisation pour optimiser les paramètres des réseaux de neurones à fonction de base radiale (RBFNN) basé sur la recherche heuristique évolutive de PSO. Ils ont divisé l'ensemble de données spambase en 70% d'ensemble d'entraînement et 30% d'ensemble de test. Les expériences sont mesurées en utilisant un autre nombre de couches cachées allant de 10 à 50. La précision obtenue était de 91,4ensemble de tests qui a conclu que l'approche hybride avait de bonnes performances par rapport à d'autres algorithmes testés sur le même jeu de données

Salah et al. [20] démontre la combinaison de l'algorithme d'optimisation de l'essaim de particules chaotiques (PSO) et de la colonie d'abeilles artificielles (ABC) pour la réduction de la dimensionnalité des caractéristiques dans le but d'améliorer la précision de la classification du spam. Les propriétés de chaque particule dans ce travail sont affichées sous forme binaire. Cela signifie qu'il a été converti en binaire à l'aide de la fonction sigmoïde. La sélection des

fonctionnalités était basée sur une fonction de fitness qui dépend de la précision obtenue avec le SVM. Les performances du système proposé ont été évaluées en tenant compte des performances du classificateur et de la dimensionnalité des vecteurs de caractéristiques sélectionnés servant d'entrées au classificateur. Cette évaluation a été réalisée à l'aide de l'ensemble de données Spam-Base, et les résultats montrent que le classificateur PSO ABC a bien fonctionné sur FS, même avec un petit échantillon de fonctionnalités sélectionnées. des travaux sur la sélection d'attributs avec des méthodes bio-inspirées[these].

Tuba et al. [21] ont utilisé l'algorithme EHO pour régler les paramètres SVM. L'approche proposée a été testée sur un ensemble de données standard et les résultats ont été obtenus à partir d'EHO et comparés à deux autres approches, à savoir GA et la méthode de recherche par grille (GRID). Des expériences informatiques ont conclu que l'algorithme EHO surpasse GA et Grid en termes de précision de classification pour le même problème de test.

Tuba et al. [22] ont utilisé l'algorithme EHO pour trouver les paramètres optimaux du SVM. Dans l'approche proposée, les paramètres SVM d'EHO ont été ajustés. Quatre expériences différentes ont été réalisées sur la base de l'ensemble de données standard. Les résultats de la simulation ont montré que les performances de la méthode proposée surpassaient les autres stratégies dans tous les cas.

Le tableau 3.1 table récapilatif des études sur la sélection des attributs à l'aide de méthodes bio-inspirées PSO.

| Auteurs | Référence | Méthode bioinspirée | Approche de FS | Domaine d'application ou noms des bases de données | Résultats Ensemble | FS |
|---|---|---|---|---|---|---|
| Liu et al. | [LIU 04] | PSO | Wrapper | Wine * | 13 | 4 |
| | | | | Ionosphere * | 34 | 8 |
| | | | | Image segmentation * | 30 | 5 |
| Wang et al. | [WAN 05] | PSO | Wrapper | Breast cancer * | 9 | 4 |
| | | | | Vote* | 16 | 8 |
| | | | | Zoo * | 16 | 5 |
| Lai et al. | [LAI 09] | PSO | Filtre | Filtrage de spams | 12800 | 46 |
| Bae et al. | [BAE 10] | PSO | Wrapper | Zoo * | 16 | 5 |
| | | | | Mushroom * | 22 | 4 |
| | | | | Breast cancer * | 9 | 4 |
| | | | | Lung * | 56 | 4 |
| Liu et al. | [LIU 11] | PSO | Wrapper | Australian * | 14 | 9 |
| | | | | Breast cancer * | 9 | 5 |
| | | | | Hill-valley * | 100 | 41 |

Tab. 3.1 : Synthse de travaux sur la slection dattributs base sur des mthodes bio-inspires

**Travaux réalisés basé sur le compte :**

Davis et al. [23], ont développé la plate-forme BotOrNot en utilisant le classificateur Random Forest comme approche de boîte noire pour comportent une réduction de la dimensionnalité et visent à évaluer si un compte Twitter est contrôlé par un humain ou une machine.

Yang et coll. [23], ont présenté une analyse empirique des tactiques d'évasion des fonctionnalités basées sur le profil et des fonctionnalités basées sur le contenu.

| References | FS techniques | Classificateur | Nbr attributs slc | Accuracy | Dataset |
|---|---|---|---|---|---|
| Yang et al. [23], 2013 | analyse empirique tactiques d'évasion de fonctionnalités basées sur le profil et tactiques d'évasion de fonctionnalités basées sur le contenu | Validation croisée 10 fois en utilisant : (forêt aléatoire, arbre de décision, réseau de Bayes et décoration) | 25 valeurs de caracté-ristiques numériques | Meilleure mesure F1 à l'aide de l'ensemble de données I : RF : 90 %, ensemble de données II : RF : 94,7 % | -Ensemble de données I : 20 000 comptes tweets indésirables, -Ensemble de données II : 35 000 Twitter comptes |
| Davis et al. [21], 2016 | Calculez le score de probabilité de bot à l'aide des MLA. | Validation croisée décuplée à l'aide de : Forêt aléatoire | 1000 valeurs de caractéristiques numériques | 95 % AUC (aire sous la courbe ROC). | -Ensemble de données de 15 000 robots sociaux vérifiés manuellement et de 16 000 comptes légitimes. |
| Rostami et Karbasi [17], 2020 | Redondance minimale Pertinence maximale algorithme [18]. | Validation croisée 10 fois en utilisant : (Forêt aléatoire, Naïf Bayes,SMV) | Test set 1: 8 Test set 2: 7 numerical | Meilleur classifier : SVM Test set 1: 98% Test set 2: 97.1% | 2 dataset de Cresci et al. [13] |

Tab. 3.2 : quelques méthodes pour la détection de faux comptes sur les OSN.

## 3.3 Filtrage des spam avec l'approche PSO

Dans [24] le cadre du filtrage de spams, proposé un travail basé sur l' approche de sélection d'attributs utilise le PSO. L'approche proposée pour la sélection d'attributs basée sur PSO a été implémentée en utilisant la version binaire de PSO (BPSO) et le classifieur de type Naïve Bayes est utilisé pour évaluer les sous ensembles obtenus par BPSO (approche Wrapper).

Le classifieur de type Naïve Bayes utilisé dans nos expérimentations a été fourni par [1] Spider Les paramètres de PSO ont été déterminés expérimentalement. Ils sont présentés dans le Tableaux (3.3), et est utilisé dans l'approche proposée.

| **Paramètres** | **Valeur** |
|---|---|
| La taille de l'essaim | Nombre de caractéristiques dans la base de données |
| W | 0.6571 |
| c1 | 1.6319 |
| c2 | 0.6239 |

Tab. 3.3 : Paramètres de PSO

[1](object-oriented machine learning package : http ://people.kyb.tuebingen.mpg.de/spider/).

Pour l'expérimentations dans le domaine du filtrage des spams, utilisé la collection Spam-Base disponible dans UCI Machine Learning Repository [2]

| Caractéristiques | Type | Explication |
|---|---|---|
| 1-48 | word_freq_WORD | Pourcentage des mots dans l'email qui sont égaux à WORD, i.e. 100 * (nombre d'occurrences du mot WORD dans l'email) / nombre total des mots dans l'email. Un mot dans ce cas est une chaîne de caractères alphanumériques délimitée par des caractères non alphanumériques ou terminée par une fin de chaîne. |
| 49-54 | char_freq_CHAR | Pourcentage de caractères dans l'email qui sont égaux à CHAR, i.e. 100 * (nombre des occurrences de caractère) / nombre total des caractères dans l'email. |
| 55 | capital_run_length _average | Moyenne des longueurs des séquences ininterrompues de lettres majuscules. |
| 56 | capital_run_length _longest, | Longueur de la séquence ininterrompue de lettres majuscules la plus longue. |
| 57 | capital_run_length _total | Somme des longueurs des séquences ininterrompues de lettres majuscules = nombre total de lettres majuscules dans l'email |

Tab. 3.4 : Description de la base SpamBase

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

SpamBase contient les informations relatives à 4601 messages, avec 1813 (39,4spams. Chaque message est décrit en fonction de 57 attributs, le 58ème correspond à la classe du message : spam ou message légitime voir Tableau (3.5).

dans [24] divisé la base Spam-Base en 2 ensembles : un ensemble d'apprentissage contenant 3250 exemples (avec 1250 spams), et un ensemble de test contenant 1351 exemples (avec 583 spams).

Afin de comparer les résultats issus de ces différents algorithmes, utiliser l'erreur d'apprentissage BER pour évaluer les performances du classifieur.

BER : Balanced Error Rate est la moyenne entre le taux d'erreur de la classe positive et celui de la classe négative. BER = (L'erreur des exemples positifs (faux positifs) + L'erreur des exemples négatifs (faux négatifs))/2.

L'erreur des exemples positifs (faux positifs) = nombre des exemples positifs mal classés/nombre total des exemples positifs.

L'erreur des exemples négatifs (faux négatifs) = nombre des exemples négatifs mal classés/nombre total des exemples négatifs.

| **Méthode** | **Erreur** | **Nombre d'attributs sélectionnés** | **Numéros des attributs sélectionnés** |
|---|---|---|---|
| Avant sélection | 0.3128 | 57 | / |
| Classique | 0.2467 | 24 | 4, 47, 38, 22, 41, 48, 32, 46, 34, 15, 44, 31, 42, 14, 39, 20, 43, 29, 33, 40, 28, 30, 35, 51 |
| PSO | 0.1545 | 16 | 2, 15, 24, 25, 26, 27, 30, 31, 32, 33, 35, 38, 42, 43, 46, 48 |

Tab. 3.5 : Comparaison entre les performances avant et après PSO pour la base Spambase

**Discutions :**

Le travail cité dans [24] est dédié pour détecter les spams on utilisons une approche basée sur PSO qui a comme rôle l'amélioration de l'auto-apprentissage par l'optimisation de nombre d'attribut utilisés. D'après le tableau (3.5) on constate que le SVM avec des attributs optimisés par PSO donne une meilleur détection que le SVM seul, Cela signifie que la PSO permet de connaître les caractéristique les plus importants en éliminant les caractéristique qui bruite la détection.

## 3.4 Conclusion

Dans le présent chapitre un état de l'art sur les approche de détection des spams et spammeurs utilisant apprentissage automatique en particulier SVM combiné avec PSO,et EHO. on a terminer ce chapitre par une discution.

# Chapitre 4

# Implémentions et Expérimentation

## 4.1 Implémentation

Dans ce chapitre nous allons mettre en œuvre une implémentation pour la détection des spammeurs dans Twitter en utilisant deux approches de classifications pour nous permettre de faire une étude comparative :

La 1 ère est avec l'algorithme de classification SVM (Support Vector Machine) est un modèle d'apprentissage automatique supervisé qui utilise des algorithmes de classification pour les problèmes de classification binaire.

La 2 ème en essaye d'utilise un algorithme hybride entre SVM et le PSO pour optimiser la performance de notre classification

### 4.1.1 Environnement

Nous avons développé notre modèle spécialisé dans la classification à l'aide de la plate forme google Colab, permet à n'importe qui d'écrire et d'exécuter le code Python de son choix par le biais du navigateur. C'est un environnement particulièrement adapté au machine learning, à l'analyse de données et l'affichage des résultats, à l'aide d'un ordinateur qu'il a les caractéristiques suivantes :

- RAM : 4 GB.
- Espace disque : 128 GB.
- CPU : Intel(R) Core(TM)i5-3470QM.
- CPU Freq : 3.20 GHz.
- Système d'exploitation : 64 bits.

Fig. 4.1 : Google colab

### 4.1.2 Fonctionnalités de détection des spammeurs

Les spammeurs peuvent être détectés à l'aide des méthodes suivantes : **faux contenu**, **fausse identification d'utilisateur** et **détection de spam basée sur l'URL**, **détection de spam dans des sujets populaires**.

Dans notre implémentation on va se concentré sur la détection de faux utilisateurs (dataset des utilisateurs twitter pour la classification).

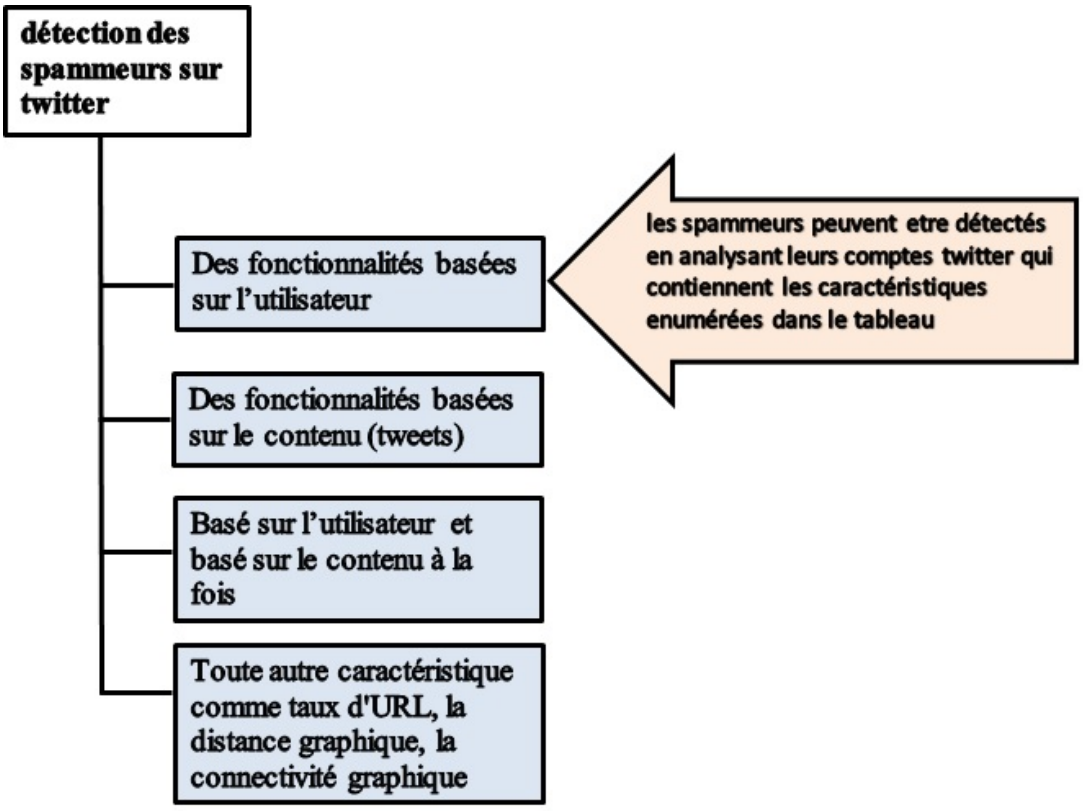


Fig. 4.2 : Taxonomie de la détection des spammeurs(faut utilisateur) sur Twitter

### 4.1.3 Description du dataset

L'expérience est réalisée on utilisons le dataset des comptes utilisateurs twitter. L'ensemble de données ce compose en deux dataset dont une pour les faux utilisateurs(fusers.csv) (spammeurs) contient 1337 comptes et la deuxième dataset pour les utilisateurs légitimes(legiuser.csv) (non-spammeurs ) contient 1481 comptes.

Dans ce tableau, contient certaines des caractéristiques courantes des comptes d'utilisateurs Twitter et leurs

type, avec une explication simple pour chaque fonctionnalité

| n° | attribus | type | description |
|---|---|---|---|
| 1 | id | int64 | La représentation entière de l'identifiant unique pour cet utilisateur. Ce nombre est supérieur à 53 bits et certains langages de programmation peuvent avoir des difficultés/défauts silencieux à l'interpréter. |
| 2 | name | object | est un identifiant personnel utilisé pour vous identifier auprès de vos amis. |
| 3 | screen_name | object | identifiant ou alias avec lequel cet utilisateur s'identifie. |
| 4 | statuses_count | int64 | Le nombre de Tweets (y compris les retweets) émis par l'utilisateur. |
| 5 | followers_count | int64 | Le nombre d'abonnés que ce compte compte actuellement. |
| 6 | friends_count | int64 | Le nombre d'utilisateurs suivis par ce compte. |
| 7 | favourites_count | int64 | Le nombre de Tweets que cet utilisateur a aimés pendant la durée de vie du compte. |
| 8 | listed_count | int64 | Le nombre de listes publiques dont cet utilisateur est membre. |
| 9 | created_at | object | La date et l'heure UTC auxquelles le compte d'utilisateur a été créé sur Twitter. |
| 10 | url | object | Une URL fournie par l'utilisateur en association avec son profil. |
| 11 | lang | object | La valeur sera définie sur null. Toujours disponible via le compte/les paramètres GET comme langue |
| 12 | time_zone | object | La valeur sera définie sur null. Toujours disponible via le compte/les paramètres GET en tant que tzinfo_name |
| 13 | location | object | L'emplacement défini par l'utilisateur pour le profil de ce compte. |
| 14 | default_profile | float64 | Lorsqu'il est vrai, indique que l'utilisateur n'a pas modifié le thème ou l'arrière-plan de son profil utilisateur. |

| | | | |
|---|---|---|---|
| 15 | default_profile_image | float64 | Lorsqu'il est vrai, indique que l'utilisateur n'a pas téléchargé sa propre image de profil et qu'une image par défaut est utilisée à la place. |
| 16 | geo_enabled | float64 | La valeur sera définie sur null. Ce champ doit être vrai pour que l'utilisateur actuel puisse joindre des données géographiques. |
| 17 | profile_image_url | object | La valeur sera définie sur null. |
| 18 | profile_banner_url | object | L'URL basée sur HTTPS pointant vers la représentation Web standard de la bannière de profil téléchargée de l'utilisateur. En ajoutant un élément de chemin final de l'URL, il est possible d'obtenir différentes tailles d'image optimisées pour des affichages spécifiques. |
| 19 | profile_use_ background_image | float64 | La valeur sera définie sur null. |
| 20 | profile_background_ image_url_https | object | La valeur sera définie sur null. |
| 21 | profile_text_color | object | La valeur sera définie sur null. |
| 22 | profile_image_url_https | object | La valeur sera définie sur null. |
| 23 | profile_sidebar_border_color | object | La valeur sera définie sur null. |
| 24 | profile_background_tile | float64 | La valeur sera définie sur null. |
| 25 | profile_sidebar_fill_color | object | La valeur sera définie sur null. |
| 26 | profile_background_image_url | object | La valeur sera définie sur null. |
| 27 | profile_background_color | object | La valeur sera définie sur null. |
| 28 | profile_link_color | object | La valeur sera définie sur null. |
| 29 | utc_offset | float64 | La valeur sera définie sur null. Toujours disponible via le compte/les paramètres GET |
| 30 | protected | float64 | Lorsqu'il est vrai, indique que cet utilisateur a choisi de protéger ses Tweets. |
| 31 | verified | float64 | Lorsqu'il est vrai, indique que l'utilisateur a un compte vérifié. Voir Comptes vérifiés . |
| 32 | description | object | l'UTF-8 définie par l'utilisateur décrivant son compte. |

Tab. 4.1 : Dictionnaire de attributs utilisateur

### 4.1.4 Stratégie

**Partie 1** : Un modèles d'apprentissage automatique utilisée basé sur la classification SVM. 75% des données sont utilisées comme ensemble d'entraînement et 25% des données restantes sont utilisées comme ensemble de test.

**Partie 2** : Dans cette partie on a appliquer le même modèle de classification (SVM) avec l'approche d'optimisation PSO pour slectionner les meilleurs attributs et lancer la classification encore une fois.

Les performances de ces modèles pour les mêmes comptes d'entrée sont mesurées à l'aide de Precision, spécifité,sencibilité et F-Measure résumés dans le tableau

### 4.1.5 Architecture

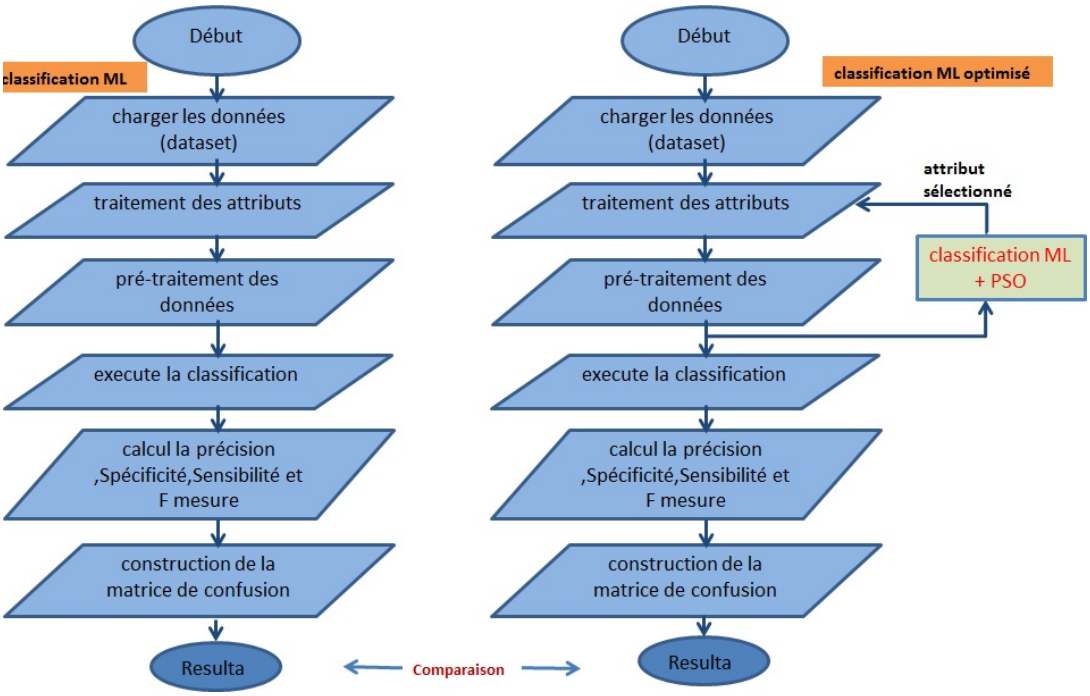


Fig. 4.3 : Architecture du système

1. **charger les bibliotèques nécessaires :**

   **- Scikit-learn :** est une bibliothèque d'apprentissage automatique gratuite pour Python, il comporte divers algorithmes comme les Naïve bayes, Random forests, et k-neighbours, et il prend également charge les bibliothèques numériques et scientifiques Python comme NumPy.

   **- NumPy :** acronyme de Numerical Python, est un package permettant d'effectuer efficacement des calculs scientifiques en Python, il a des capacités de généré des nombres aléatoires, des fonctions d'algèbre linéaire de base et encore plus.

- **Pandas :** est un outil d'analyse et de manipulation des données open source, rapide, puissant, flexible et facile à utiliser, construit pour le langage de programmation Python.

- **PySwarms :** est une boîte à outils de recherche extensible pour l'optimisation des essaims de particules (PSO) en Python. Il est destiné a l'intelligence en essaim qui préfèrent une interface déclarative de haut niveau pour implémenter PSO dans leurs problèmes.

.

2. télécharger les dataset et afficher quelque exemple :

   Dans notre projet, nous avons utilisé deux groupes de base de données (data set) se nommé (users) contient la base de donné des utilisateurs légitimes (1481 users) et (fusers) contient la base de donné des faux utilisateurs –spammeur- (1337), en tant que fichier CSV, avec 34 attributs disposées comme suit figure 3.3.


Fig. 4.4 : télécharger les dataset.

‘

3. Suppression des attributs inutiles :

   Les spammeurs peuvent être détectés en analysant leurs compte twitter qui contiennent les caractéristiques énumères dans le tableau. Comme certaines de ces caractéristiques,

telle que ”id”,”name”,”screen name”,……,et”description” sont contrôlées par l’utilisateur, elles sont inutiles pour la détection des faux comptes (spammeur)figure 3.4 :


Fig. 4.5 : suppression des colonnes inutiles.

4. Création d’un ensemble de données :

   Après la Conversion de DataFrame en tableau, Vérification de la disponibilité de **URL** et **Time Zone** et Convertir les données en **float64**,en fusionne à partir d’un ensemble de données de profil légitime et un autre de faux a un seul ensemble de données figure3.4


Fig. 4.6 : Création d’un ensemble de données.

5. Fractionnement de l’ensemble de données en train et test :

   !

## Fractionnement de l'ensemble de données en train et test

```
X_train, X_test, y_train, y_test = train_test_split( X, Y,
    test_size=0.25, random_state=40)
```

```
[18] print("Shape of X_train:", X_train.shape)
     print("Shape of y_train:", y_train.shape)
     print("Shape of x_test:", X_test.shape)
     print("Shape of y_test:", y_test.shape)
```

```
Shape of X_train: (2113, 13)
Shape of y_train: (2113,)
Shape of x_test: (705, 13)
Shape of y_test: (705,)
```

Fig. 4.7 : Fractionnement de l'ensemble de données en train et test.

**Partie 1:**

**Appliquer le classificateur SVM :**

On a choisis SVM puisque il est une classe d'algorithmes d'apprentissage initialement définis pour la discrimination c'est-à-dire la prévision d'une variable qualitative binaire.

## classification

```
[20] # exécuter le modèle SVM

     for model in [ SVC(random_state=40)]:

         print("[INFO]: Fitting", str(model), "...")

         model.fit(X_train, y_train)

         y_pred = model.predict(X_test)

         evaluate(y_pred, y_test)
```

```
[INFO]: Fitting SVC(random_state=40) ...
[INFO]: Accuracy: 96.17
[INFO]: F1 Score: 95.95
[INFO]: Specificity: 99.07
[INFO]: Sensitivity: 93.02
```

Fig. 4.8 : Résultat de classificateur SVM.

On obtient une accuracy de :**96.17%**

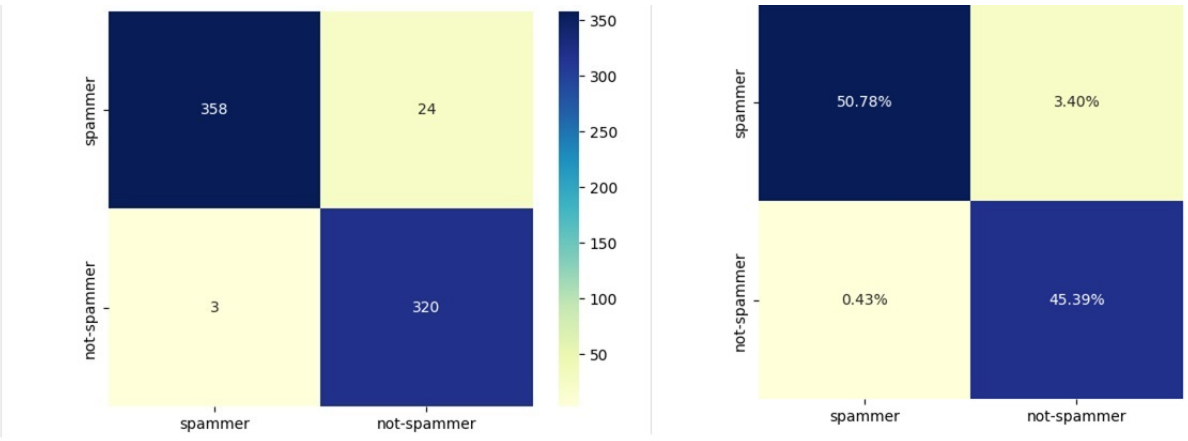


Fig. 4.9 : Appliquer le classificateur SVM.

**Partie 2:**

Dans cette partie, on a appliqué un système de sélection de caractéristiques pour minimiser la dimensionnalité et augmenter la performance du système dans les mêmes paramètres et condition basé sur **PSO**.

Cela se fait en intégrant PySwarms (module PSO) après l'étape de prétraitement des données, afin de réduire le nombre des attribut et pour exploiter les résultats, on lance la classification avec les mêmes condition et même classificateur (SVM) avec l'élimination des attributs sort du l'exécution PSO pour nous permettre comparer les résultats.

**pyswarms pour la selection des attributs :**

```
- INFO - Optimize for 100 iters with {'c1': 0.5, 'c2': 0.5, 'w': 0.9, 'k': 13, 'p': 2}
best_cost=0.0214
- INFO - Optimization finished | best cost: 0.021434159061277692, best pos: [1 1 1 1 1 1 1 1 1 1 0 1 1]
```

Fig. 4.10 : Appliquer le classificateur SVM+PSO.

l'interprétation du résultat obtenu par PSO nous donne les attributs a éliminer :

```
Class_numbers=np.array(['statuses_count', 'followers_count', 'friends_count',
       'favourites_count', 'listed_count', 'url', 'time_zone',
       'default_profile', 'default_profile_image', 'profile_background_tile',
       'utc_offset', 'protected', 'verified'])

print("Colonnes eliminés avec PSO:\n")

for x, y in zip(Class_numbers, pos_justpso):
    if y == 0:
        print(x)
        columns_just_pso.append(x)
```

```
Colonnes eliminés avec PSO:

utc_offset
```

Fig. 4.11 : Attributs éliminés avec SVM+PSO.

Dans ce tableau, nous pouvons voir que, généralement, l'attribut 11 (time zone) n'est pas sélectionné dans les meilleurs sous-ensembles d'attributs. En analysant ce tableau, il est clair que après l'élimination de cette attribut par le PSO et de relancer la classification encore une fois avec SVM,donne un meilleur résultat en termes de précision de la classification

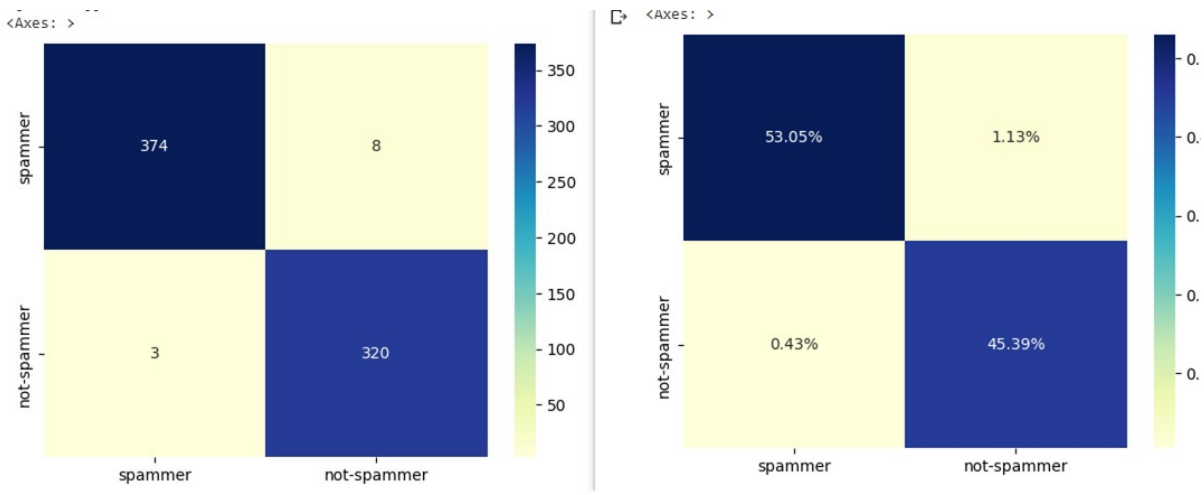


Fig. 4.12 : Appliquer le classificateur SVM+PSO.

## 4.2 Expérimentation

Dans cette partie, deux techniques, à savoir, SVM et SVM optimisé (avec l'approche PSO) sont utilisées pour détecter ou classifier les spammeurs sur Twitter. Les expériences sont menées à l'aide de dataset faux et légitimes utilisateurs accessibles au public. Les performances de ces techniques sont mesurées à l'aide de différentes métriques Accuracy, Précision, Rappel et F-Mesure. Les résultats montrent que SVM-PSO fonctionnent bien par rapport à SVM.

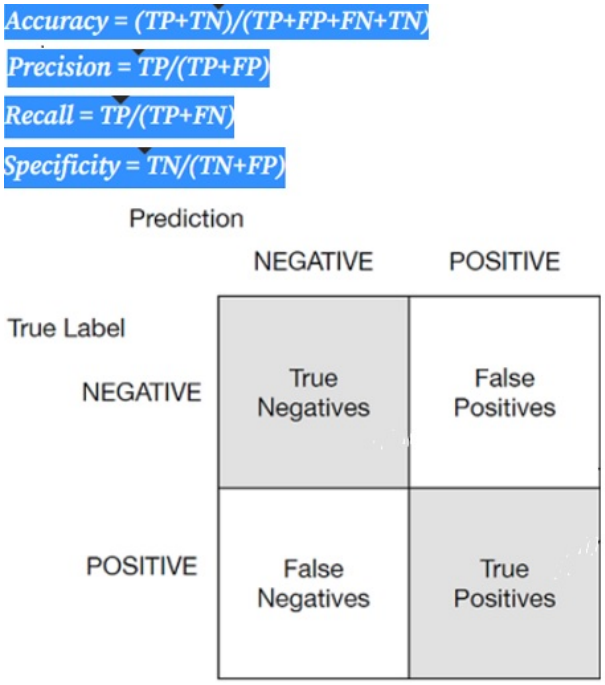


Fig. 4.13 : Les différentes valeurs d'évaluation.

Le rappel est aussi appelé sensibilité. Precision st aussi appelé spécificité

| technique de détection des spammeurs sur twitter | | accuracy | f-measure | Specificité | Sensitivité | spammeur | non spammeur |
|---|---|---|---|---|---|---|---|
| classification ML (avec SVM) | spammeur | 96.17 | 95.95 | 99.07 | 93.02 | 50.78 | 3.40 |
| | non spammeur | | | | | 0.43 | 45.39 |
| classification optimisé (SVM+PSO) | spammeur | 98.44 | 98.31 | 99.07 | 97.56 | 53.05 | 1.13 |
| | non spammeur | | | | | 0.43 | 45.39 |

Tab. 4.2 : Analyse des performances des classificateurs après l'optimisation pour la détection des spammeurs sur twitter.

### 4.2.1 Discutions des résultats :

On remarque que les résultats obtenus dans le domaine de detection des spammeurs sont les meme que ceux obtenus précédemment citer dans l'etat de l'art.

la question qui se pose est ce que on remplaçant PSO par EHO on va obtenir une amélioration meilleur? mais pour le moment on a pas encore implémenter notre modèle en se basant sur EHO,sachant qu'on peut dire que ça va donner des résultats meilleurs vu que EHO approuvé leur fiabilité dans des travaux relatifs a d'autre domaine d'application que le sien par-rapport au PSO,et cela reste a prouver par expérimentation.

# Conclusion Générale

La détection des spammeurs était et reste aujourd'hui et demain le défi de sécurité des réseau sociaux en particulier twitter, plusieurs approches étais proposés mais les résultats demande une grande amélioration et optimisation. Sachant que les méta-heuristiques est parmi les outils d'optimisation qui ont prouvé leur efficacité et parmi les célèbres méthodes ceux inspirés de la nature,les plus connus on peut citer PSO et EHO.

Le but de ce travail est de faire une comparaison entre deux algorithme de détection des spammeurs dans Twitter ,proposition de classification des faux utilisateurs basés sur deux approche bio inspiré PSO vs EHO.

Malheureusement y'a peu de sources qui expose EHO ce qui nous amené à comparer entre un algorithmes d'apprentissage automatique SVM et PSO – SVM (SVM optmisé) comme suite :

- SVM (support vector machine) atteint une précision de **96.17%**.

- SVM (support vector machine) et PSO atteint une précision de **99%**, donc la sélection d'attributs basée sur PSO donne, généralement, de meilleurs résultats par rapport SVM à la détection des spammeurs en termes de précision de classification.

# Références

[1] M. Verma and S. Sofat, "Techniques to detect spammers in twitter-a survey," *International Journal of Computer Applications*, vol. 85, no. 10, 2014.

[2] G. Stringhini, C. Kruegel, and G. Vigna, "Detecting spammers on social networks," in *Proceedings of the 26th annual computer security applications conference*, pp. 1–9, 2010.

[3] D. Irani, M. Balduzzi, D. Balzarotti, E. Kirda, and C. Pu, "Reverse social engineering attacks in online social networks," in *International conference on detection of intrusions and malware, and vulnerability assessment*, pp. 55–74, Springer, 2011.

[4] A. A. Amleshwaram, N. Reddy, S. Yadav, G. Gu, and C. Yang, "Cats : Characterizing automation of twitter spammers," in *2013 Fifth International Conference on Communication Systems and Networks (COMSNETS)*, pp. 1–10, IEEE, 2013.

[5] F. Masood, A. Almogren, A. Abbas, H. A. Khattak, I. U. Din, M. Guizani, and M. Zuair, "Spammer detection and fake user identification on social networks," *IEEE Access*, vol. 7, pp. 68140–68152, 2019.

[6] B. Mukunthan and M. Arunkrishna, "Spam detection and spammer behaviour analysis in twitter using content based filtering approach," in *Journal of Physics : Conference Series*, vol. 1817, p. 012014, IOP Publishing, 2021.

[7] X. Zheng, Z. Zeng, Z. Chen, Y. Yu, and C. Rong, "Detecting spammers on social networks," *Neurocomputing*, vol. 159, pp. 27–34, 2015.

[8] S. Kodati, K. P. Reddy, S. Mekala, P. S. Murthy, and P. C. S. Reddy, "Detection of fake profiles on twitter using hybrid svm algorithm," in *E3S Web of Conferences*, vol. 309, p. 01046, EDP Sciences, 2021.

[9] A. El Dor, *Perfectionnement des algorithmes d'optimisation par essaim particulaire : applications en segmentation d'images et en électronique*. PhD thesis, Paris Est, 2012.

[10] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, vol. 4, pp. 1942–1948, IEEE, 1995.

[11] O. Y. Assaf and N. M. Jassam, "An enhanced particle swarm optimization algorithm for e-mail spam filtering,"

[12] S. Singh and A. K. Singh, "Detection of spam using particle swarm optimisation in feature selection.," *Pertanika Journal of Science & Technology*, vol. 26, no. 3, 2018.

[13] I. Macedo, "Implementing the particle swarm optimization (pso) algorithm in python," *Retrieved from Medium : https ://medium. com/analytics-vidhya/implementing-particleswarm-optimization-pso-algorithm-in-python-9efc2eb179a6*, 2018.

[14] S. Almufti, R. Asaad, and B. Salim, "Review on elephant herding optimization algorithm performance in solving optimization problems," *International Journal of Engineering & Technology*, vol. 7, pp. 6109–6114, 2018.

[15] G.-G. Wang, S. Deb, and L. d. S. Coelho, "Elephant herding optimization," in *2015 3rd international symposium on computational and business intelligence (ISCBI)*, pp. 1–5, IEEE, 2015.

[16] J. Li, H. Lei, A. H. Alavi, and G.-G. Wang, "Elephant herding optimization : variants, hybrids, and applications," *Mathematics*, vol. 8, no. 9, p. 1415, 2020.

[17] K. Liu, Y. Tan, and X. He, "Particle swarm optimization based learning method for process neural networks," in *Advances in Neural Networks-ISNN 2010: 7th International Symposium on Neural Networks, ISNN 2010, Shanghai, China, June 6-9, 2010, Proceedings, Part I 7*, pp. 280–287, Springer, 2010.

[18] D. Ye, Z. Chen, and J. Liao, "A new algorithm for minimum attribute reduction based on binary particle swarm optimization with vaccination," in *Advances in Knowledge Discovery and Data Mining : 11th Pacific-Asia Conference, PAKDD 2007, Nanjing, China, May 22-25, 2007. Proceedings 11*, pp. 1029–1036, Springer, 2007.

[19] M. A. M. Foqaha, "Email spam classification using hybrid approach of rbf neural network and particle swarm optimization," *International Journal of Network Security & Its Applications*, vol. 8, no. 4, pp. 17–28, 2016.

[20] H. M. Saleh, "An efficient feature selection algorithm for the spam email classification," *Periodicals of Engineering and Natural Sciences*, vol. 9, no. 3, pp. 520–531, 2021.

[21] E. Tuba and Z. Stanimirovic, "Elephant herding optimization algorithm for support vector machine parameters tuning," in *2017 9th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pp. 1–4, IEEE, 2017.

[22] E. Tuba, I. Ribic, R. Capor-Hrosik, and M. Tuba, "Support vector machine optimized by elephant herding algorithm for erythemato-squamous diseases detection," *Procedia computer science*, vol. 122, pp. 916–923, 2017.

[23] A. Aboud, N. Rokbani, S. Mirjalili, A. M. Qahtani, F. S. Alharithi, O. Almutiry, A. Hussain, and A. M. Alimi, "A quantum beta distributed multi-objective particle swarm optimization algorithm for twitter fake accounts detection," 2022.

[24] K. Menghour, *Approches Bio-inspirées pour la Sélection d'Attributs*. PhD thesis, THESE Présentée en vue de l'obtention du diplôme de Doctorat 3ème Cycle …, 2014.